

13th International Workshop  
on Plasma Edge Theory  
South Lake Tahoe, 2011/09/21

# Integrated modeling of core, edge and peripheral plasmas

Atsushi FUKUYAMA

*Department of Nuclear Engineering, Kyoto University*

M. Miki, H. Seto, T. Ikari,

*Kyoto University*

M. Honda, N. Hayashi, K. Shimizu, T. Takizuka

*Japan Atomic Energy Agency*



# OUTLINE

---

1. Integrated modeling of toroidal plasmas
2. Integrated modeling activities
3. Integrated modeling code TASK
4. Interface for integrated modeling
5. Various level of transport modeling
6. 1D modeling of core, edge, and peripheral plasmas
7. 1D core + 2D peripheral coupling (Yagi: next talk)
8. 1D modeling coupled with MHD stability and pellet
9. Summary

# Integrated modeling of toroidal plasmas

---

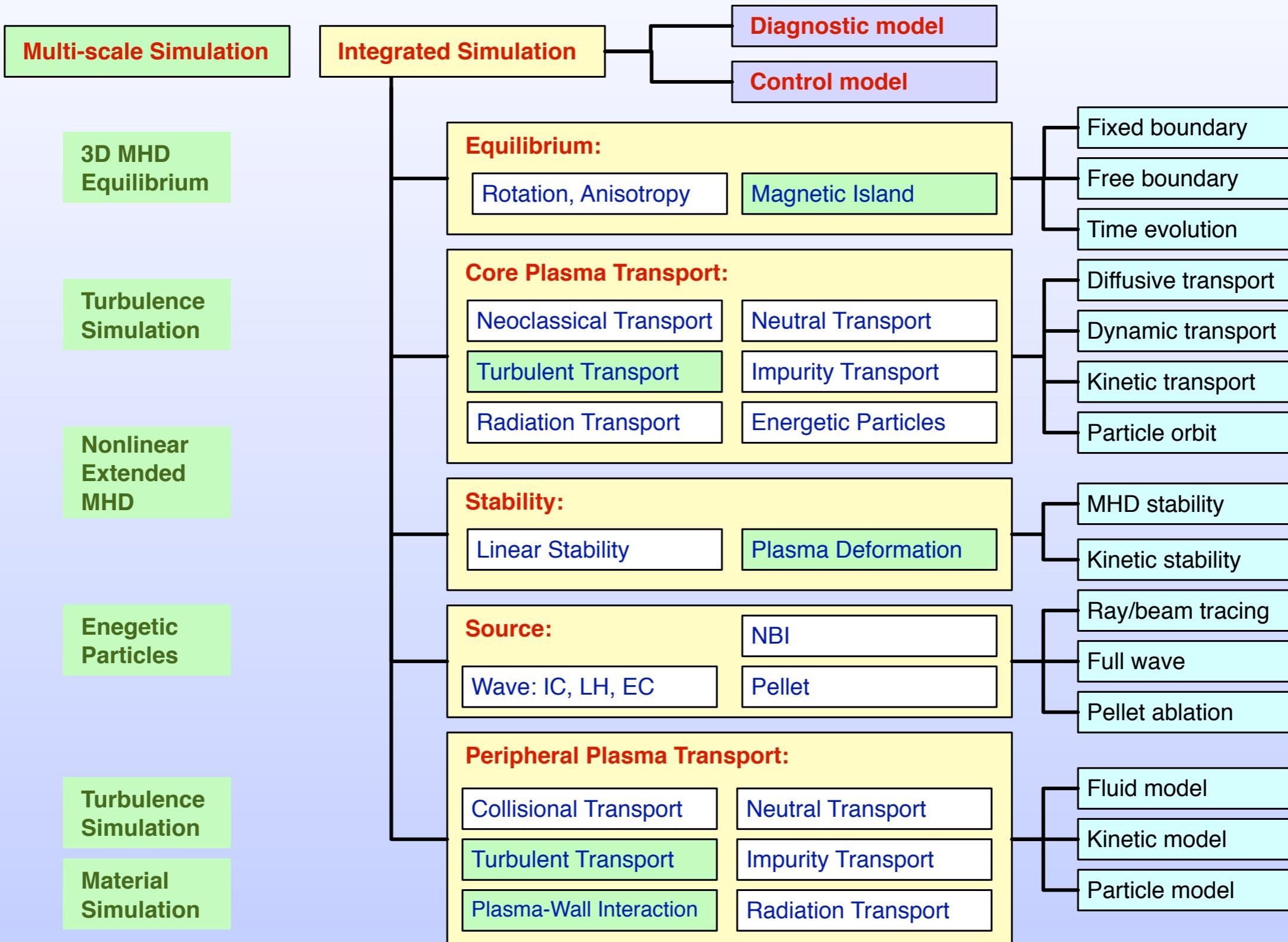
- \* In order to
  - ▶ predict the performance of future fusion devices,
  - ▶ optimize their operation scenario,
  - ▶ design DEMO fusion reactors
- \* we need reliable tool describing
  - ▶ whole plasma including core, edge, scrape-off layer (SOL), and divertor plasmas as well as plasma-wall interactions
  - ▶ whole discharge period including startup, sustainment, probabilistic incidents, and shut down
- \* Wide range of time scale, spatial scale, and understanding
  - ▶ Integrated simulation of various component codes
  - ▶ Various levels of physics model

# Integrated modeling activities

---

- \* US: SciDAC, FSP (Fusion Simulation Program)
  - ▶ FACETS: Framework Application for Core-Edge Transport Simulations
  - ▶ CPES: Center for Plasma Edge Simulation
  - ▶ SWIM: Simulation of Wave Interactions with Magnetohydrodynamics
- \* EU: ITM TF
  - ▶ Integrated Tokamak Modeling - Task Force
  - ▶ data model: CPO (Consistent Physical Objects)
  - ▶ code interface: UAL (Universal Access Layer)
- \* JA: BPSI
  - ▶ Burning Plasma Simulation Initiative
  - ▶ data structure and data interface: BPSD
- \* ITER: IMEG
  - ▶ IMAS: Integrated Modelling Analysis Suits
  - ▶ IM standards and guideline

# Integrated simulation



# BPSI: Burning Plasma Simulation Initiative

---

## Research Collaboration of Universities, NIFS and JAEA

- Targets of BPSI
  - Framework for collaboration of various plasma simulation codes
    - Common interface for data transfer and execution control
    - Standard data set for data transfer and data storage
    - Reference core code: TASK
    - Helical configuration: included
  - Physics integration with different time and space scales
    - Transport during and after a transient MHD events
    - Transport in the presence of magnetic islands
    - Core-SOL interface and ...
  - Advanced technique of computer science
    - Parallel computing: PC cluster, Scalar-Parallel, Vector-Parallel
    - Distributed computing: GRID computing, Globus, ITBL

# Integrated Modeling Activity in the Framework of BPSI

---

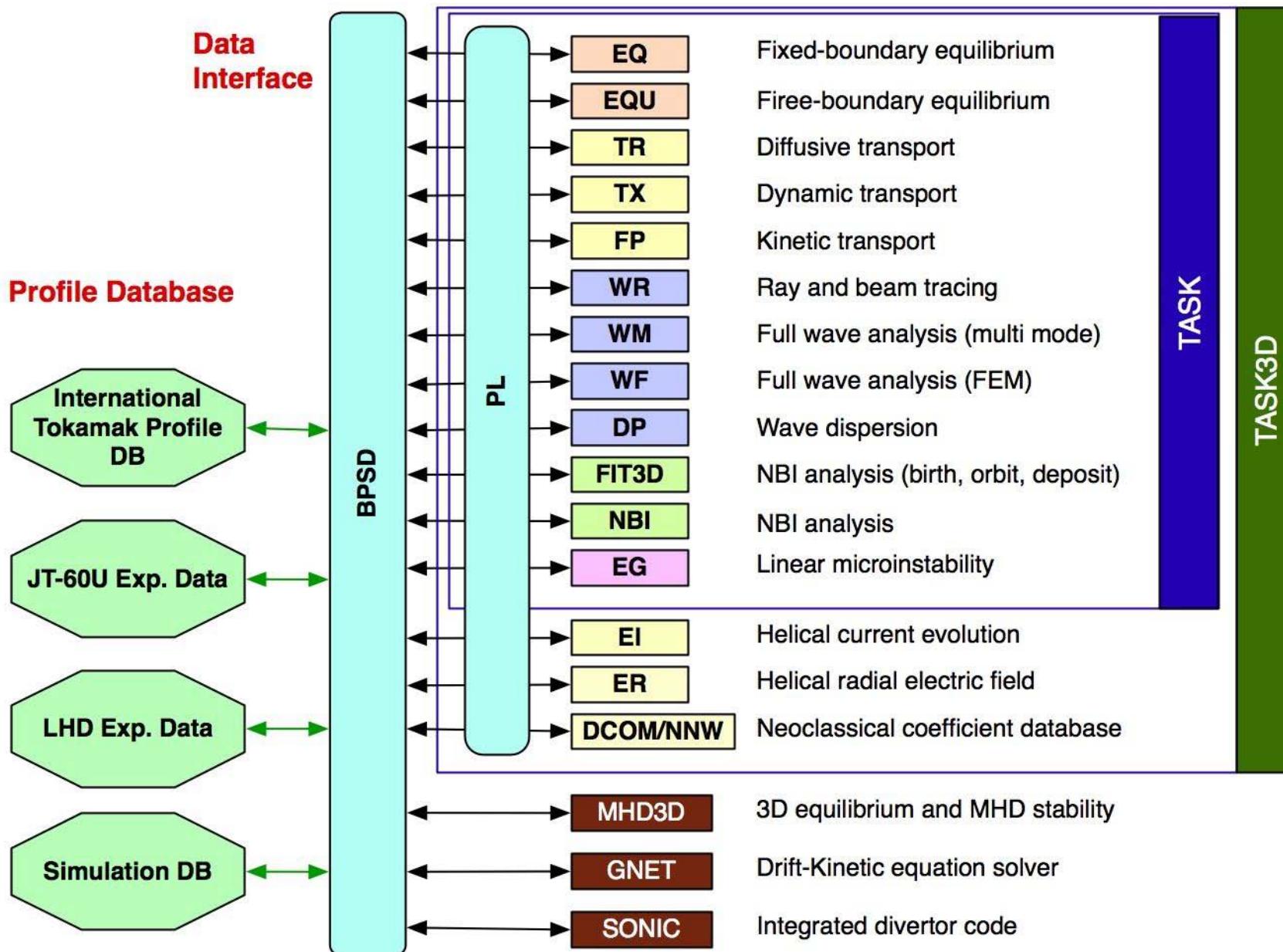
- **Development of Integrated Code**
  - **TASK**: Tokamak and others (Kyoto U)
  - **TOPICS-IB + SONIC + MARG2D**: Tokamak (JAEA)
  - **TASK3D**: Helical system LHD and 3D tokamak (NIFS and Kyoto U)
- **Exchange of Components**
  - **TOPICS+TASK/WM**: Simulation of ITER operation
  - **TASK+SONIC**: Coupling of core and peripheral plasmas
  - **TASK+MARG2D**: ELM modeling
- **Extension**
  - **TASK for ST**: Spherical tokamak (QUEST, LATE, ...)
  - **TASK for RFP**: Reversed Field Pinch (with Nihon U)
  - **TASK for RT**: Ring trap (with Tokyo U)

# Integrated Modeling Code: TASK

---

- **Transport Analysing System for TokamaK**
- **Features**
  - **Core of Integrated Modeling Code in BPSI**
    - Modular structure for easier maintenance
    - Reference implementation of standard data set and interface
  - **Various Heating and Current Drive Scheme**
    - EC, LH, IC, AW, NB
  - **High Portability**
    - Most of library routines included
    - Original graphic libraries (X11, Postscript, OpenGL, SVG)
  - **Development using CVS** (Version control for collaboration)
  - **Open Source:** <http://bpsi.nucleng.kyoto-u.ac.jp/task/>
  - **Parallel Processing using MPI and PETSc**
- Developed in **Kyoto University** since 1993

# Present Structure of TASK and TASK3D



# Component Collaboration

---

- **Role of Component Interface**
  - **Data exchange between components:** **BPSD**
    - **Standard dataset:** Specify set of data
    - **Specification of data exchange interface:** initialize, set, get
    - **Specification of file i/o interface:** save, load
  - **Execution control:** **BPSX**
    - **Specification of execution control interface:** initialize, setup, exec, visualize, terminate
    - **Uniform user interface:** parameter input, graphic output
- **Role of data exchange interface:** **TASK/PL**
  - **Keep present status of plasma and device**
  - **Store history of plasma**
  - **Interface to experimental data base**

# Policies of BPSD

---

- **Minimum and Sufficient Dataset**
  - To minimize the data to be exchanged
  - Mainly profile data
  - Routines to calculate global integrated quantities
    - separately provided
- **Minimum Arguments in Interfaces**
  - To maximize flexibility
  - Use derived data type or struct
  - Only one dataset in the arguments of an interface
- **Minimum Kinds of Interfaces**
  - To make modular programming easier
  - Use function overloading
- **Language:** Fortran95, Fortran2003

# Data Exchange Interface: BPSD

---

- **Standard dataset:** Specify data to be stored and exchanged
  - **Data structure:** Derived type (Fortran95): structured type

	time	<code>plasmaf%time</code>
	number of grid	<code>plasmaf%nrmax</code>
	number of species	<code>plasmaf%nsamax</code>
e.g.	normalized radius	<code>plasmaf%rho(nr)</code>
	Species specifier	<code>plasmaf%ns(nsa)</code>
	plasma density	<code>plasmaf%data(nr,nsa)%density</code>
	plasma temperature	<code>plasmaf%data(nr,nsa)%temperature</code>

- **Specification of API:**

- **Program interface**

	<b>Set data</b>	<code>bpsd_set_data(plasmaf,ierr)</code>
	<b>Get data</b>	<code>bpsd_get_data(plasmaf,ierr)</code>
e.g.	<b>Save data to file</b>	<code>bpsd_save(ierr)</code>
	<b>Load data from file</b>	<code>bpsd_load(ierr)</code>

- **BPSD data file** (`bpsddata`): Binary file of all existing bpsd data

# BPSD Standard Dataset

Category	Name	EQ	TR	TX	FP	WR	WM	DP
Shot data	<b>bpsd_shot_type</b>	—	—	—	—	—	—	—
Device data	<b>bpsd_device_type</b>	in	in	in	in			
1D equilibrium data	<b>bpsd_equ1D_type</b>	out	in	in	in			
2D equilibrium data	<b>bpsd_equ2D_type</b>	out			in	in	in	in
1D metric data	<b>bpsd_metric1D_type</b>	out	in	in	in			
2D metric data	<b>bpsd_metric2D_type</b>	out			in	in	in	in
Plasma species data	<b>bpsd_species_type</b>	in	in	in	in			in
Fluid plasma data	<b>bpsd_plasmaf_type</b>	in	out	out	i/o			in
Kinetic plasma data	<b>bpsd_plasmak_type</b>				out			in
Transport matrix data	<b>bpsd_trmatrix_type</b>		i/o					
Transport source data	<b>bpsd_trsource_type</b>		i/o	i/o	i/o	out	out	
Dielectric tensor data	<b>bpsd_dielectric_type</b>					in	in	out
Full wave field data	<b>bpsd_wavef_type</b>				in	out		
Ray tracing field data	<b>bpsd_waver_type</b>				in		out	
Beam tracing field data	<b>bpsd_waveb_type</b>				in		out	
User defined data	<b>bpsd_0/1/2ddata_type</b>	—	—	—	—	—	—	—

# BPSD: Standard dataset

---

- **Public dataset**: data exchange
  - **Standard dataset**
    - Without dataName: **bpsd\_device\_type**, **bpsd\_equ1D\_type**, etc.
    - With dataName:   **bpsd\_trmatrix\_type**: neoclassical, turbulence, ...  
                       **bpsd\_trsource\_type**:  $P_{EC}$ ,  $P_{NB}$ ,  $S_{NB}$ , ...
  - **User defined dataset**
    - **bpsd\_data0D\_type**, **bpsd\_data1D\_type**,  
**bpsd\_data2D\_type**, **bpsd\_data3D\_type**
- **Internal dataset**: unaccessible from users
  - **bpsd\_shotx\_type**: text data including shotID
  - **bpsd\_data0Dx\_type**: no profile
  - **bpsd\_data1Dx\_type**: 1D profile ( $\rho$ : normalized radius)
  - **bpsd\_data2Dx\_type**: 2D profile ( $\rho, \chi$ : poloidal angle)
  - **bpsd\_data3Dx\_type**: 3D profile ( $\rho, \chi, \zeta$ : toroidal angle)

## Example of internal dataset structure: **data1Dx**

---

```
type bpsd_data1Dx_type
    integer :: status = 0! 0:unallocated 1:data-allocated 2:data-assigned
                           ! 3:spline-allocated 4:spline-created
    integer :: nrmax      ! Number of radial points
    integer :: ndmax      ! Number of data
    integer :: idum        ! Dummy
    real(rkind) :: time
    real(rkind), dimension(:), pointer :: rho
    real(rkind), dimension(:, :), pointer :: data
    real(rkind), dimension(:, :, :), pointer :: spline
    character(len=32) :: dataName
    character(len=8)  :: created_date
    character(len=10) :: created_time
    character(len=5)  :: created_timezone
    character(len=9)  :: dummy
    character(len=32), dimension(:), pointer :: kid    ! item name
    character(len=32), dimension(:), pointer :: kunit ! unit of item
end type bpsd_data1Dx_type
```

# Example of data structure: plasmaf

---

```
type bpsd_plasmaf_data
    real(kind=rkind) :: pn      ! Number density [m^-3]
    real(kind=rkind) :: pt      ! Temperature [eV]
    real(kind=rkind) :: pptr    ! Parallel temperature [eV]
    real(kind=rkind) :: ptpp    ! Perpendicular temperature [eV]
    real(kind=rkind) :: pu      ! Parallel flow velocity [m/s]
end type bpsd_plasmaf_data

type bpsd_plasmaf_type
    real(kind=rkind) :: time
    integer :: nrmax      ! Number of radial points
    integer :: nsamax     ! Number of particle species
    integer, dimension(:) :: ns_nsa   ! Species specifier
    real(kind=rkind), dimension(:), allocatable :: rho  ! norm. radius
    real(kind=rkind), dimension(:), allocatable :: qinv ! 1/q
    type(bpsd_plasmaf_data), dimension(:, :, :), allocatable :: data
end type bpsd_plasmaf_type
```

# BPSD Code Interface

---

- **bpsd\_set\_data(data,ierr):**
  - Copy data into internal dataset
- **bpsd\_get\_data(data,ierr):**
  - Copy of interpolate data fram internal dataset
  - If nrmax=0, copy data; otherwise interpolate for given mesh.
- **bpsd\_save(ierr):**
  - Save all BPSD data into a file
  - Name of the file is optional.
- **bpsd\_load(ierr):**
  - Load all BPSD data from a file
  - Name of the file is optional.
- **Interfaces for history archivng are under consideration.**

# Examples of sequence in a component

---

- **TR\_EXEC(dt)**: Transport time evolution

```
call bpsd_get_data(plasmaf,ierr)
call bpsd_get_data(metric1D,ierr)
local data <- plasmaf,metric1D
advance time step dt
plasmaf <- local data
call bpsd_set_data(plasmaf,ierr)
```

- **EQ\_CALC**: Equilibrium calculation

```
call bpsd_get_data(plasmaf,ierr)
local data <- plasmaf
calculate equilibrium
update plasmaf
call bpsd_set_data(plasmaf,ierr)
equ1D,metric1D <- local data
call bpsd_set_data(equ1D,ierr)
call bpsd_set_data(metric1D,ierr)
```

# Standard data for SOL/divertor plasmas

---

— proposed by Professor Hatayama (Keio U) —

- **Plasma data**
  - MHD equilibrium data
  - Wall structure/material data
  - Plasma data at the Core / SOL boundary (e.g.,  $n_s$ ,  $T_s$ ,  $cdots$ )
- **Atomic & Molecular data**
  - Ionization/recombination/excitation rate coefficient/cross section
  - Radiation loss function
- **PWI data**
  - Particle and Energy Reflection Coefficient at the wall
  - Sputtering yield, Emission energy, Emission angle

# Several Approaches on Workflow

---

- **Monolithic code approach:** original approach
  - **Memory-based data exchange**
  - **Template:** call `bpsd_get_data`  
`calculation`  
`call bpsd_set_data`
- **Command approach:** for script and workflow
  - **File-based data exchange**
  - **Template:** call `bpsd_load ← bpsddata`  
`call bpsd_get_data`  
`calculation`  
`call bpsd_set_data`  
`call bpsd_save → bpsddata`
- **Pre- and post- process approach:** no modification of the code
  - **pre-process:** `bpsddata`  $\implies$  input file
  - **post-process:** output file  $\implies$  `bpsddata`

# Levels of Transport Modeling

- **Fluid model**

Diffusive transport equation:  $n(\rho, t), v_\phi(\rho, t), T(\rho, t)$

**TR**

Dynamic transport equation:  $n(\rho, t), \mathbf{u}(\rho, r), T(\rho, t)$

**TX**

- **Kinetic model**

Bounce-averaged drift-kinetic equation:  $f(p, \theta_p, \rho, t)$

**FP**

Axisymmetric gyrokinetic equation:  $f(p, \theta_p, \rho, \chi, t)$

**XGC0**

Gyrokinetic equation:  $f(p, \theta_p, \rho, \chi, \zeta, t)$

**GT5D, GKV**

Full kinetic equation:  $f(p, \theta_p, \phi_g, \rho, \chi, \zeta, t)$

**PARASOL**

# 1D Dynamic Transport Code: TASK/TX

---

- **Dynamic Transport Equations** (TASK/TX) *M. Honda et al, JCP (2008)*
  - A set of flux-surface averaged equations
  - Two fluid equations for electrons and ions
    - Continuity equations
    - Equations of motion (radial, poloidal and toroidal)
    - Energy transport equations
  - Maxwell's equations
  - Diffusion equations for three-group neutrals
- **Self-consistent description of plasma rotation and electric field**
  - Neoclassical transport: driven by poloidal viscosity
  - Quasi-neutrality is not assumed.
- **Extention to include 3D effects**
  - Helical neoclassical viscosity force :  $F_{s\theta}^{\text{HNC}}$ ,  $F_{s\phi}^{\text{HNC}}$
  - Diffusion due to destroyed magnetic surfaces :  $D_m v_{Ts}$

# Transport Equation in TASK/TX (1)

---

- **Continuity equations:**

$$\frac{\partial n_s}{\partial t} = -\frac{1}{r} \frac{\partial}{\partial r} (r n_s u_{sr}) + \frac{1}{r} \frac{\partial}{\partial r} \left( r D_m v_{Ts} \frac{\partial n_s}{\partial r} \right) + S_s$$

- **Equations of motion:**

$$\frac{\partial}{\partial t} (m_s n_s u_{sr}) = -\frac{1}{r} \frac{\partial}{\partial r} (r m_s n_s u_{sr}^2) + \frac{1}{r} r m_s n_s u_{s\theta}^2 - \frac{\partial}{\partial r} (n_s T_s) + e_s n_s (E_r + u_{s\theta} B_\phi - u_{s\phi} B_\theta)$$

$$\begin{aligned} \frac{\partial}{\partial t} (m_s n_s u_{s\theta}) = & -\frac{1}{r^2} \frac{\partial}{\partial r} (r^2 m_s n_s u_{sr} u_{s\theta}) + \frac{1}{r^2} \frac{\partial}{\partial r} \left[ r^3 m_s n_s \mu_s \frac{\partial}{\partial r} \left( \frac{u_{s\theta}}{r} \right) \right] + e_s n_s (E_\theta - u_{sr} B_\phi) \\ & + \frac{1}{r} \frac{\partial}{\partial r} \left[ r D_m v_{Ts} \frac{\partial}{\partial r} (m_s n_s u_{s\theta}) \right] + F_{s\theta}^{NC} + F_{s\theta}^{HNC} + F_{s\theta}^C + F_{s\theta}^W + F_{s\theta}^L + F_{s\theta}^N + F_{s\theta}^{CX} \end{aligned}$$

$$\begin{aligned} \frac{\partial}{\partial t} (m_s n_s u_{s\phi}) = & -\frac{1}{r} \frac{\partial}{\partial r} (r m_s n_s u_{sr} u_{s\phi}) + \frac{1}{r} \frac{\partial}{\partial r} \left( r m_s n_s \mu_s \frac{\partial u_{s\phi}}{\partial r} \right) + e_s n_s (E_\phi + u_{sr} B_\theta) \\ & + \frac{1}{r} \frac{\partial}{\partial r} \left[ r D_m v_{Ts} \frac{\partial}{\partial r} (m_s n_s u_{s\phi}) \right] + F_{s\phi}^{HNC} + F_{s\phi}^C + F_{s\phi}^W + F_{s\phi}^L + F_{s\phi}^N + F_{s\phi}^{CX} \end{aligned}$$

# Transport Equation in TASK/TX (2)

---

- Heat transport equations:

$$\begin{aligned}\frac{\partial}{\partial t} \left( \frac{3}{2} n_s T_s \right) = & - \frac{1}{r} \frac{\partial}{\partial r} \left( \frac{5}{2} r u_{sr} n_s T_s - \frac{3}{2} r n_s \chi_s \frac{\partial T_s}{\partial r} \right) + e_s n_s (E_\theta u_{s\theta} + E_\phi u_{s\phi}) \\ & + \frac{1}{r} \frac{\partial}{\partial r} \left[ r D_m v_{Ts} \frac{\partial}{\partial r} (n_s t_s) \right] + P_s^C + P_s^L + P_s^R + P_s^{RF}\end{aligned}$$

- Maxwell's equation

$$\frac{1}{R} \frac{\partial}{\partial R} (R E_r) = \frac{1}{\epsilon_0} \sum_s e_s n_s$$

$$\frac{1}{c^2} \frac{\partial E_\theta}{\partial t} = - \frac{\partial B_\phi}{\partial r} - \mu_0 \sum_s e_s n_s u_{s\theta}$$

$$\frac{1}{c^2} \frac{\partial E_\phi}{\partial t} = \frac{1}{r} \frac{\partial}{\partial r} (r B_\theta) - \mu_0 \sum_s e_s n_s u_{s\phi}$$

$$\frac{\partial B_\theta}{\partial t} = \frac{\partial E_\phi}{\partial r}, \quad \frac{\partial B_\phi}{\partial t} = - \frac{1}{r} \frac{\partial}{\partial r} (r E_\theta)$$

# Transport Modeling

---

- **Neoclassical transport**
  - Poloidal and toroidal viscosity
    - ⇒ radial diffusion, resistivity, and bootstrap current
- **Turbulent transport**
  - **Particle diffusion**
    - Poloidal momentum exchange between electrons and ions
    - Intrinsic ambipolar flux (electron particle flux = ion particle flux)

$$F_{e\theta}^W = -F_{i\theta}^W = -\frac{e^2 B_\phi^2 D_{TB}}{T_e} n_e \left( u_{e\theta} - \frac{B_\theta}{B_\phi} u_{e\phi} \right)$$

- **Perpendicular viscosity:** Non-ambipolar particle flux

$$\mu_i$$

- **Thermal diffusivity:**

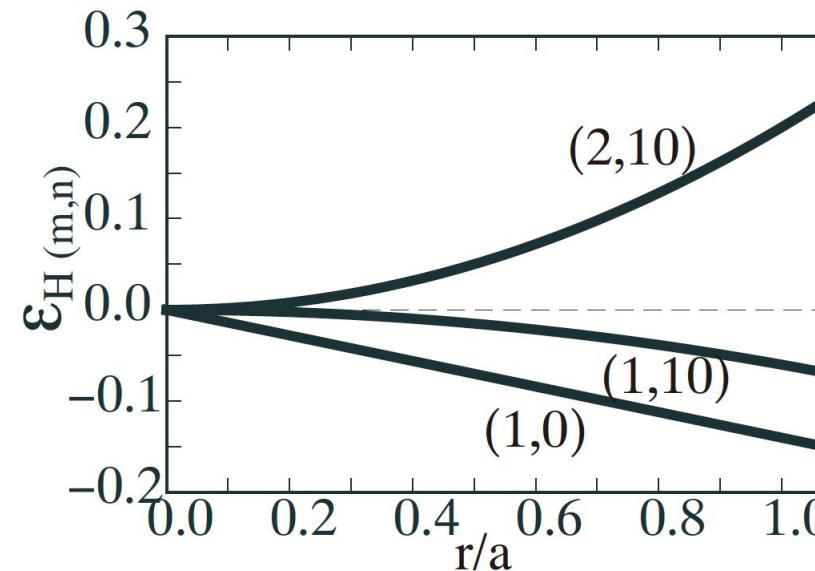
$$\chi_e, \quad \chi_i$$

# Helical magnetic field

---

- **Helical magnetic perturbation :**

$$B(r, \theta, \phi) \approx B_0 \left\{ 1 - \epsilon_T \cos \theta - \sum \epsilon_{H(m,n)} \cos(n\theta - m\phi) \right\}$$



- **Effects on radial transport**
  - Increase of **neoclassical diffusion** in the low collision regime
  - Increase of radial transport due to **destroyed magnetic surface** (ergodic region) near the plasma edge

# Neoclassical transport in helical systems

- **Neoclassical diffusion coefficient in helical plasmas**

Helically-trapped particles contribute to radial transport in low  $\nu$  regime:

- $1/\nu$  **regime**, when  $E_r$  is small
- **Collisionless detrapping regime**, when  $E_r$  is large

- **Neoclassical viscosity force**:  $F_s^{\text{HNC}}$  in TASK/TX

$$\begin{pmatrix} F_{s\theta}^{\text{HNC}} \\ F_{s\phi}^{\text{HNC}} \end{pmatrix} = -m_s n_s v_s^{\text{HNC}} \begin{pmatrix} \alpha_\theta^2 & -\alpha_\theta \alpha_\phi \\ -\alpha_\theta \alpha_\phi & \alpha_\phi^2 \end{pmatrix} \begin{pmatrix} u_{s\theta} \\ u_{s\phi} \end{pmatrix}$$

$$\alpha_\theta = \frac{l/r}{\sqrt{(l/r)^2 + (m/R)^2}}, \quad \alpha_\phi = \frac{m/R}{\sqrt{(l/r)^2 + (m/R)^2}}$$

$$v_s^{\text{HNC}} = \frac{v_{Ts} \epsilon_H^{3/2}}{R \nu_{ei}} \frac{1}{3 + 1.67 \frac{\epsilon_T \epsilon_{H,\max} \omega_E^2}{\nu_s^2}},$$

$$\omega_E = \frac{E_r}{Br} : E \times B \text{ drift frequency}$$

# Diffusion due to magnetic braiding

---

- **Magnetic braiding** due to radial fluctuation of magnetic lines near the plasma edge, give rise to **radial diffusion** of particles
- For weakly collisional plasma:

$$D_{rs} \equiv \frac{\langle (\Delta r)^2 \rangle}{\tau} = \begin{cases} D_m v_{Ts} & (\rho_{\min} < \rho < \rho_{\max}) \\ 0 & (\text{otherwise}) \end{cases}$$

$\lambda = \tau v_{Ts}$  : mean free path,  $v_{Ts}$  : thermal velocity

$$\begin{aligned} D_m(\rho) &\equiv \frac{\langle (\Delta r)^2 \rangle}{\lambda} : \text{spatial diffusion coefficient} \\ &= D_{m0} \quad (D_{m0} : \text{given}) \end{aligned}$$

# Modeling of SOL Plasma

---

- **Parallel losses in the SOL**
  - **Particle, momentum and ion heat losses: convection**

$$\nu_L = \frac{k_L C_s}{2\pi q R} \quad (a < r < b)$$

- **Electron heat loss: conduction**

$$\nu_L = k_L \frac{\chi_{||}}{(2\pi q R)^2} = k_L \frac{\kappa_0 T_e^{5/2}}{n_e (2\pi q R)^2} \quad (a < r < b)$$

- **Particle source**

$$S_e = n_0 \langle \sigma_{\text{ion}} v \rangle n_e - \nu_L (n_e - n_{e,\text{div}})$$

- **Recycling from divertor**
  - Recycling ratio:  $\Gamma_0 = 0.8$
- **Gas puff from wall**

# Determination of the radial electric field

---

- Two methods have been used to determine the radial electric field  $E_r$ 
  - **Algebraic ambipolarity relation** : radial ambipolar particle flux  $\Gamma_s^a$

$$\varepsilon_0 \frac{\partial}{\partial t} \left[ \left( 1 + \frac{c^2}{v_A^2} \right) E_r \right] = - \sum_s e_s \Gamma_s^a$$

- **Differential ambipolarity relation**: diffusion of electric field

$$\varepsilon_0 \frac{\partial}{\partial t} \left[ \left( 1 + \frac{c^2}{v_A^2} \right) E_r \right] = - \sum_s e_s \Gamma_s^a + \frac{1}{V'(r)} \frac{\partial}{\partial r} \left[ V'(r) \left( \sum_s \frac{e_s}{e} D_{E_s} \right) \frac{\partial E_r}{\partial r} \right]$$

- Corresponding equation can be derived from the equations of motion in TASK/TX.

$$\varepsilon_0 \frac{\partial}{\partial t} \left[ \left( 1 + \frac{c^2}{v_A^2} \right) E_r \right] = - \sum_s \frac{B_\phi}{B^2} \left\{ \frac{1}{r^2} \frac{\partial}{\partial r} \left[ r^3 m_s n_s \mu_s \frac{\partial}{\partial r} \left( \frac{u_{s\theta}}{r} \right) \right] + F_{s\theta}^{\text{NC}} + F_{s\theta}^{\text{HNC}} \right\}$$

$$+ \sum_s \frac{B_\theta}{B^2} \left\{ \frac{1}{r} \frac{\partial}{\partial r} \left( r m_s n_s \mu_s \frac{\partial u_{s\phi}}{\partial r} \right) + F_{s\phi}^{\text{HNC}} \right\}$$

- More physical modeling of radial electric field

# Numerical simulation

---

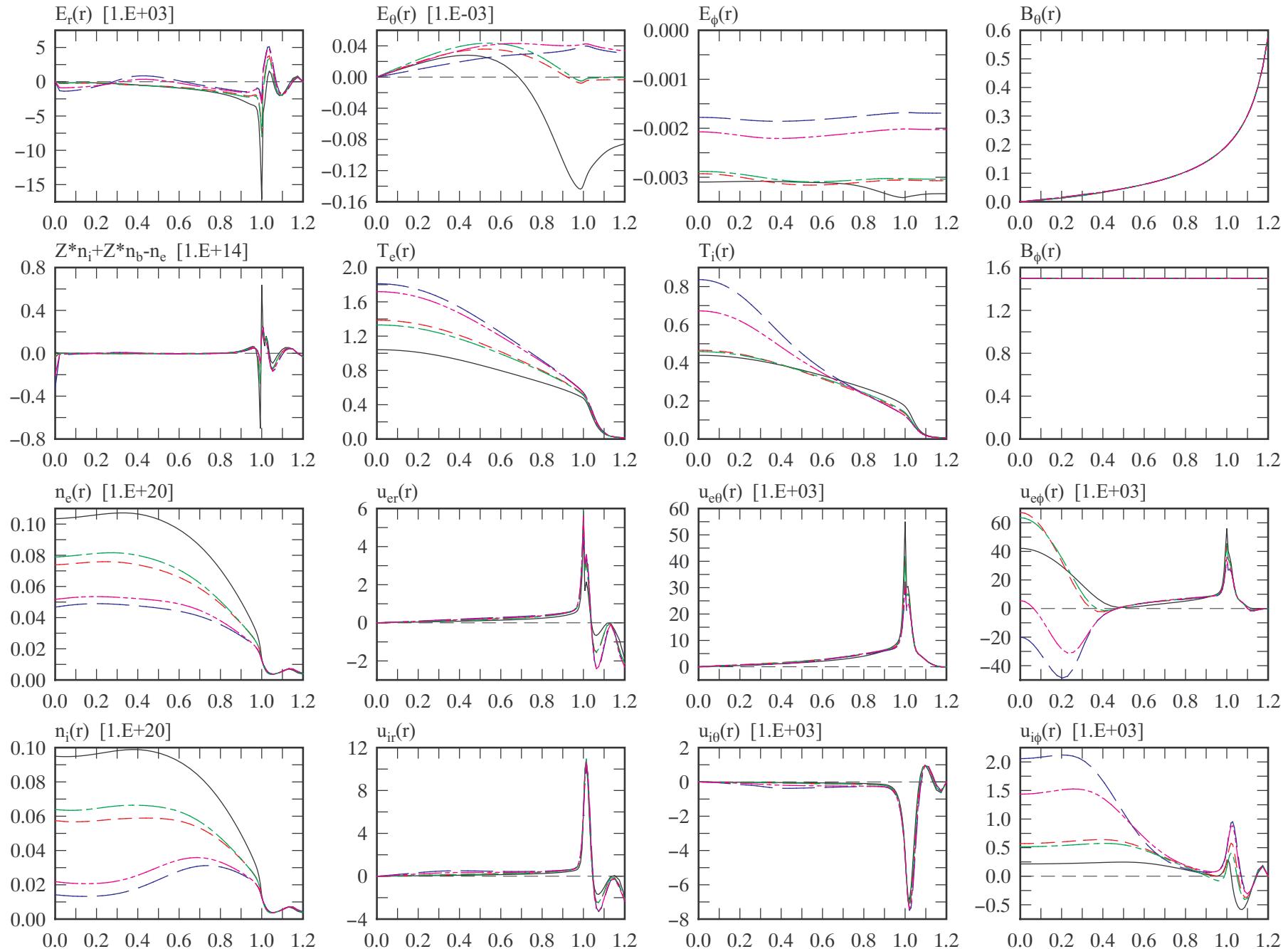
- **Plasma transport simulations with ion and electron heating**
- We used mainly typical parameters of **Large Helical Device**:

$$R = 3.7 \text{ m}, \quad a = 0.60 \text{ m}, \quad b = 0.72 \text{ m},$$

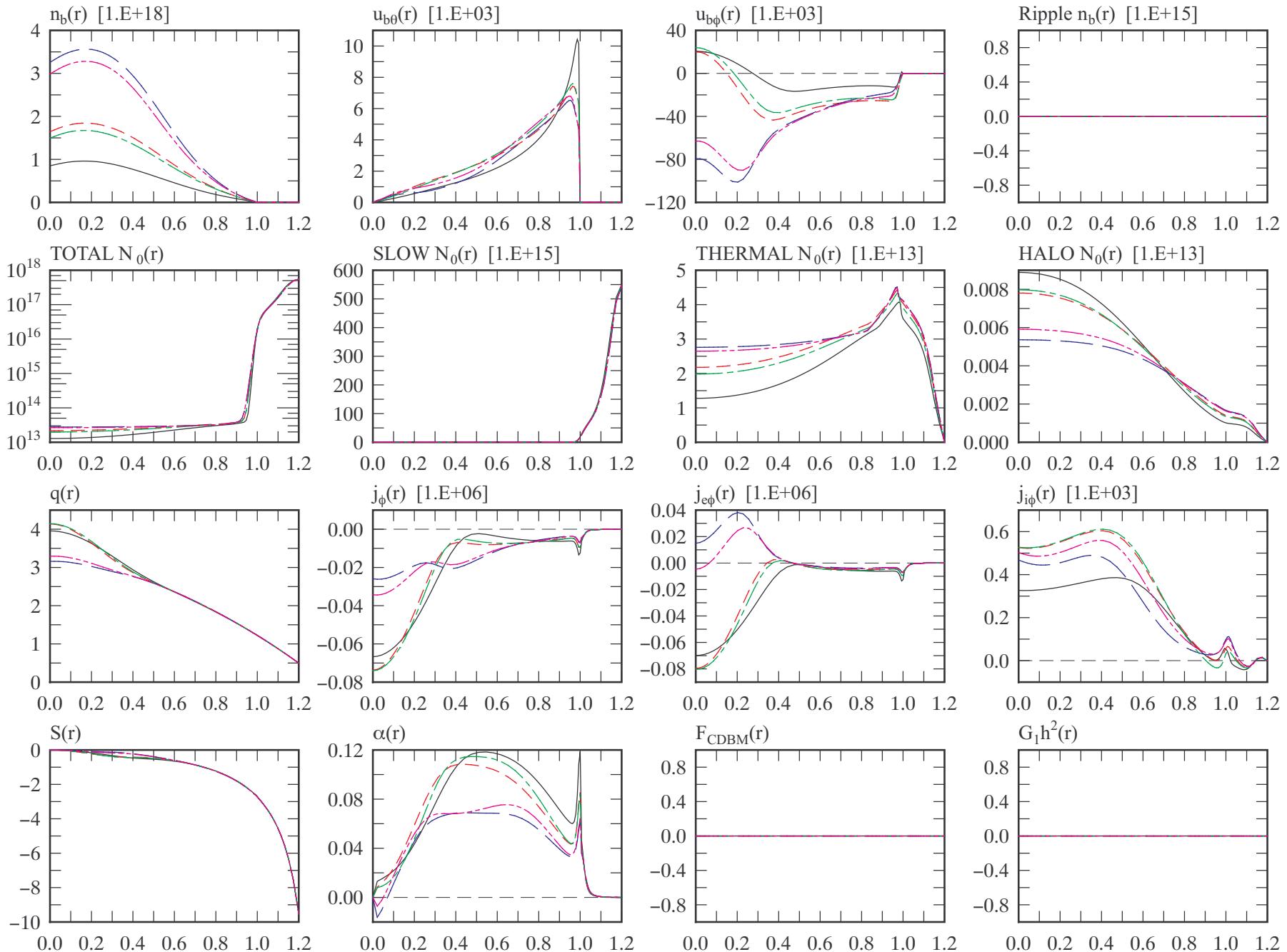
$$l = 2, \quad m = 10$$

- **Turbulent transport coefficients**:
  - **Fixed profiles** are used in the present analysis.
  - Particle diffusivity :  $D(\rho) = D(0) + [D(1) - D(0)]\rho^3$   
$$D(0) = 0.025 \text{ m}^2/\text{s}, \quad D(1) = 0.025 \text{ m}^2/\text{s}$$
  - Thermal diffusivity :  $\chi_s(\rho) = \chi_s(0) + [\chi_s(1) - \chi_s(0)]\rho^2$   
$$\chi_s(0) = 7.5 \text{ m}^2/\text{s}, \quad \chi_s(1) = 15 \text{ m}^2/\text{s}$$
  - Perpendicular viscosity :  $\mu_s(\rho) = \chi_s(\rho)$

# Typical profiles (1)

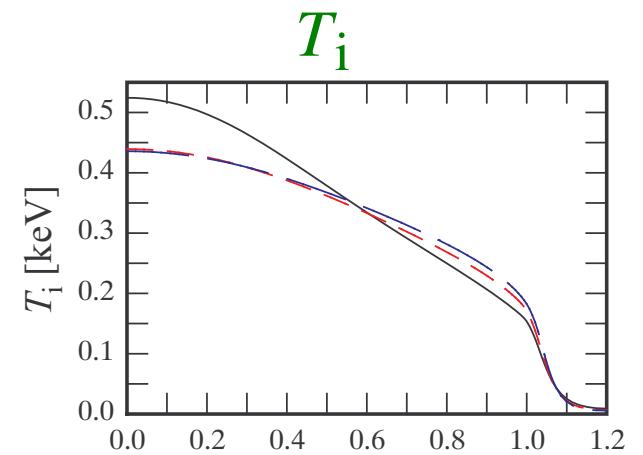
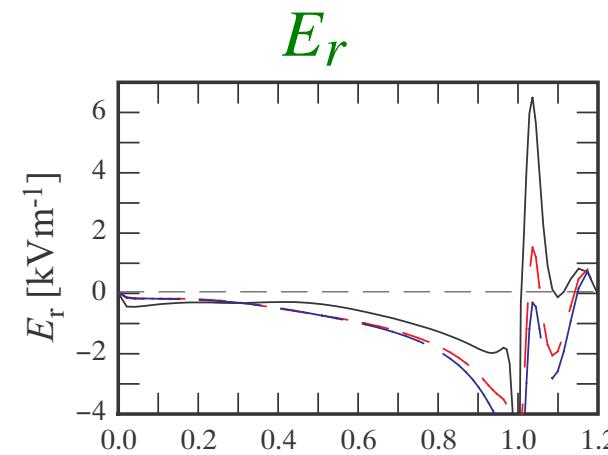
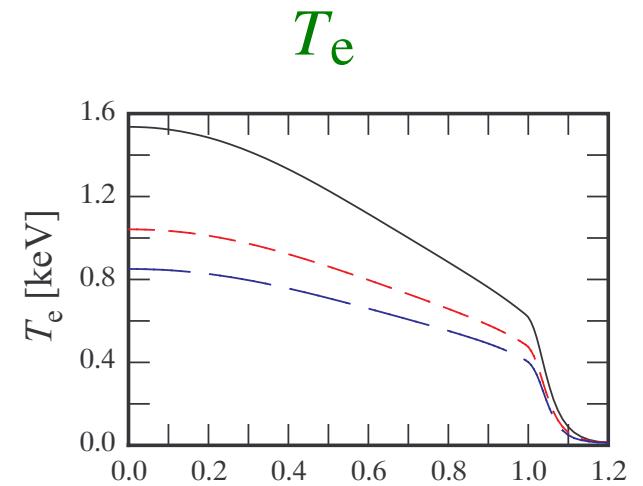
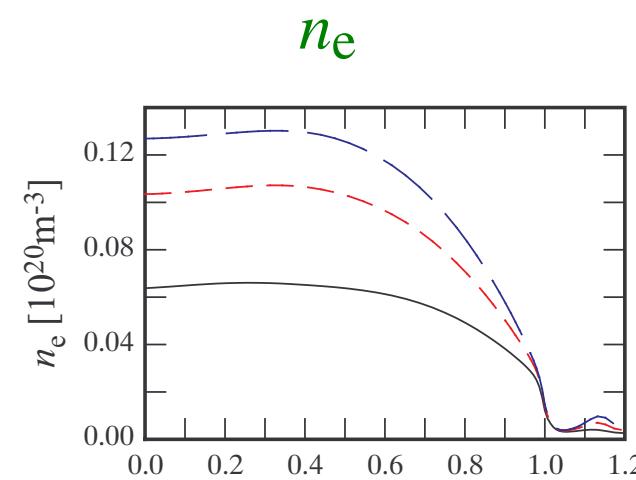
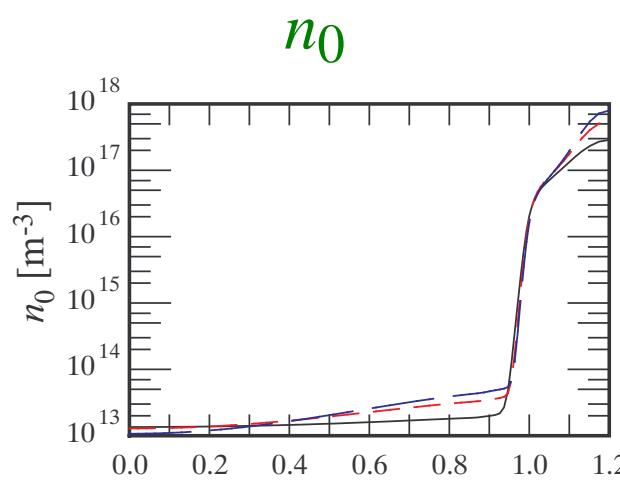


# Typical profiles (2)



# Dependence on Gas Puff Rate

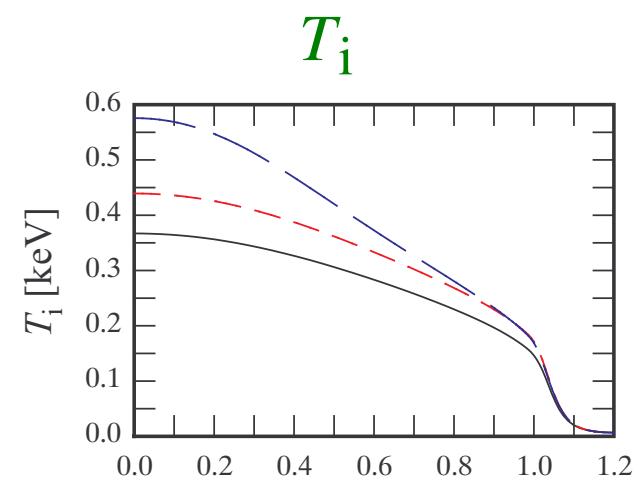
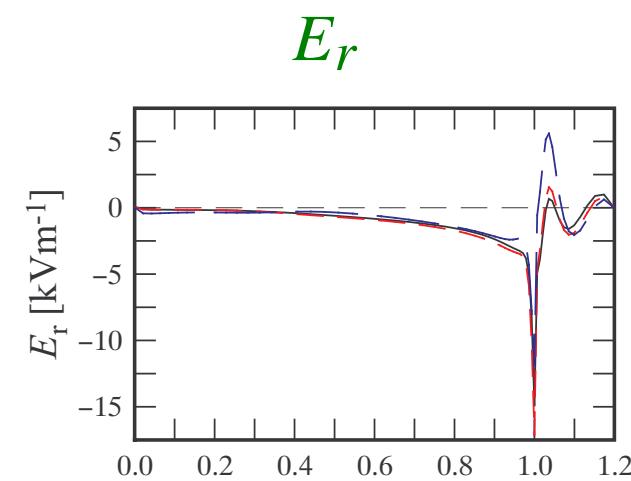
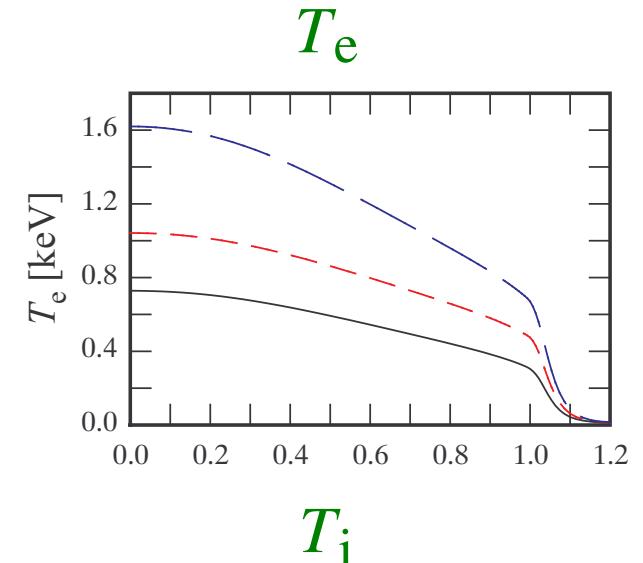
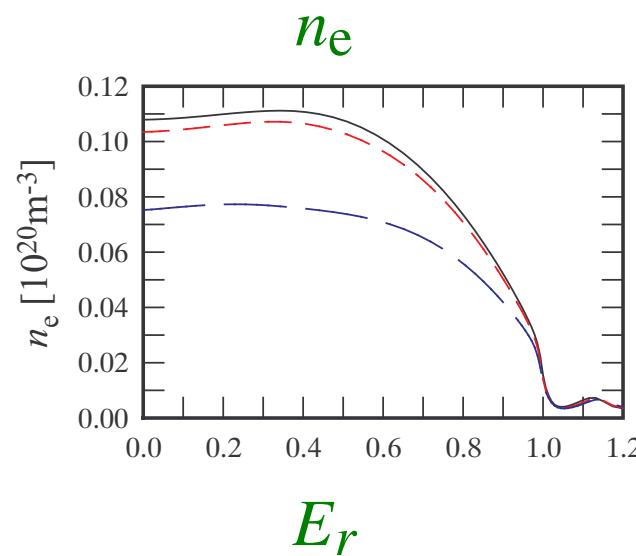
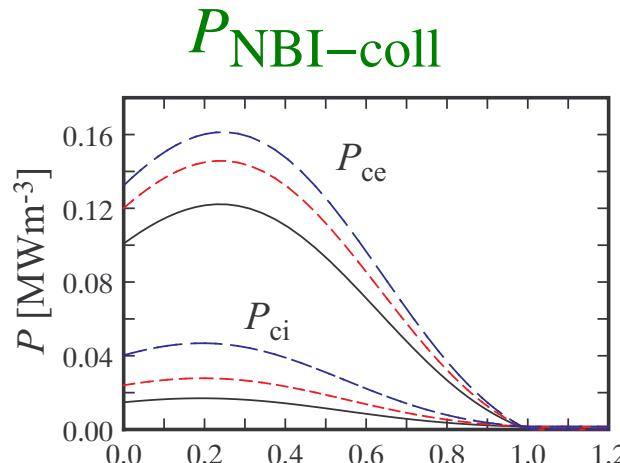
- **Gas puff rate:**  $\Gamma_{\text{gas}} = 2.5, 5.0, 7.5 \cdot 10^{20} \text{ m}^{-2} \text{s}^{-1}$



- Gas puff rate affect the plasma density and the plasma temperature.

# Dependence on NBI Heating Power

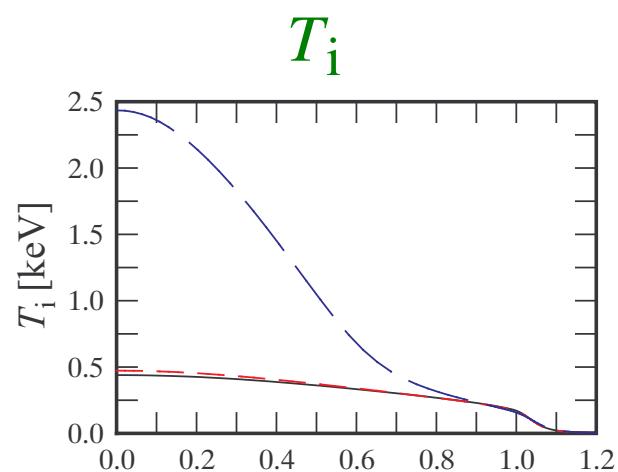
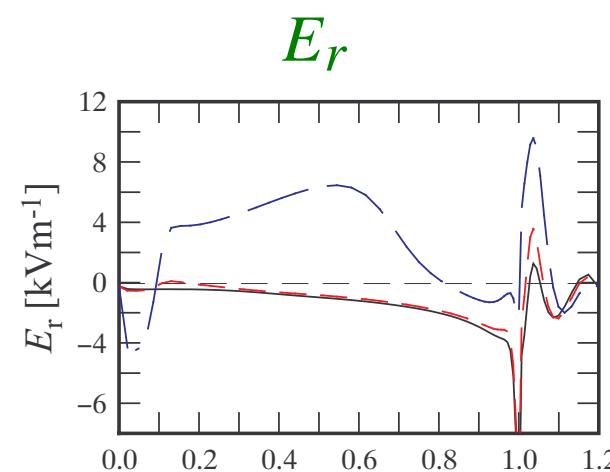
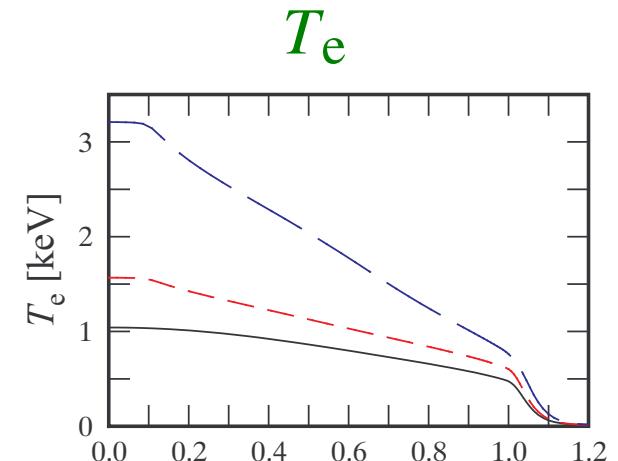
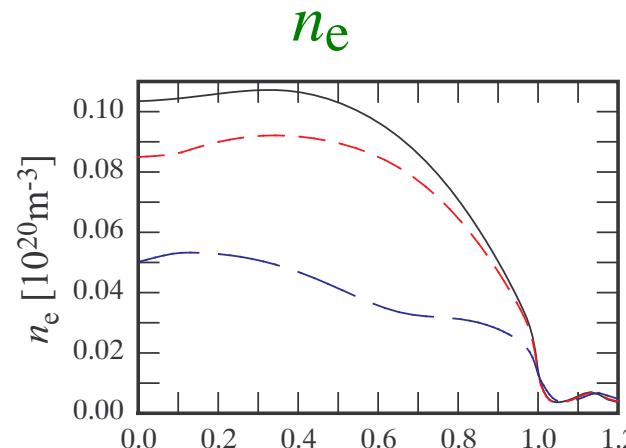
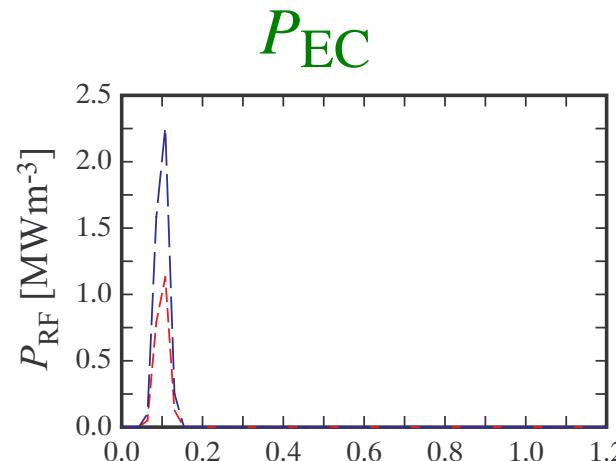
- **NBI heating power:**  $P_{\text{NBI}} = 1.6, 2.0, 2.4 \text{ MW}$



- NBI heating increases the plasma temperature and reduces the plasma density.

# Dependence on EC Heating Power

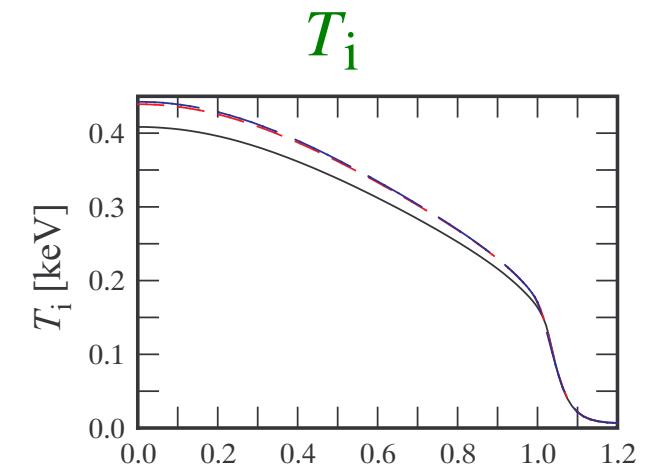
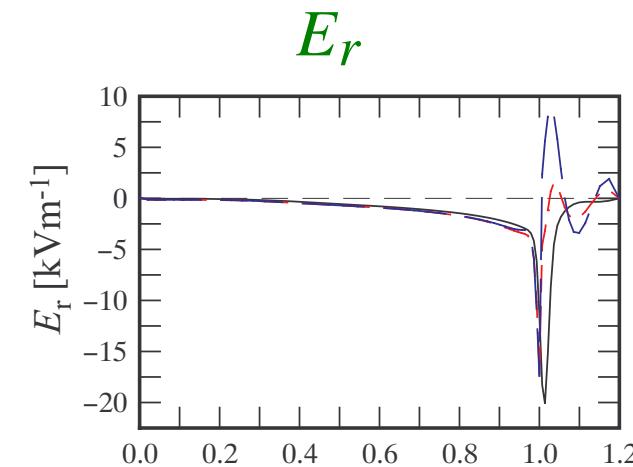
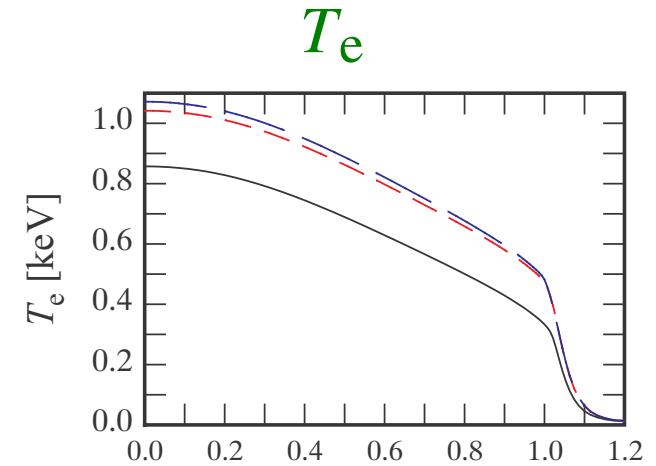
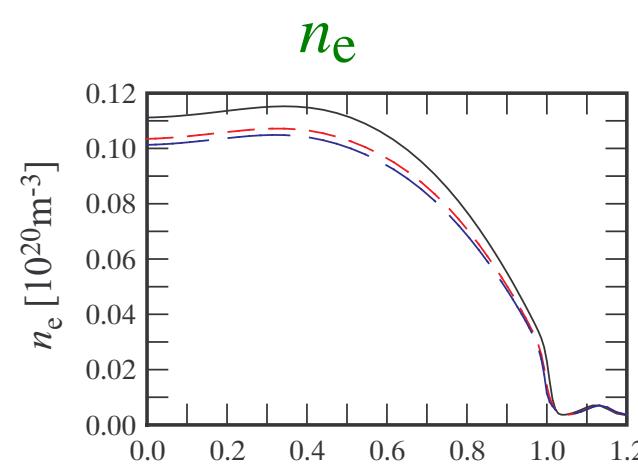
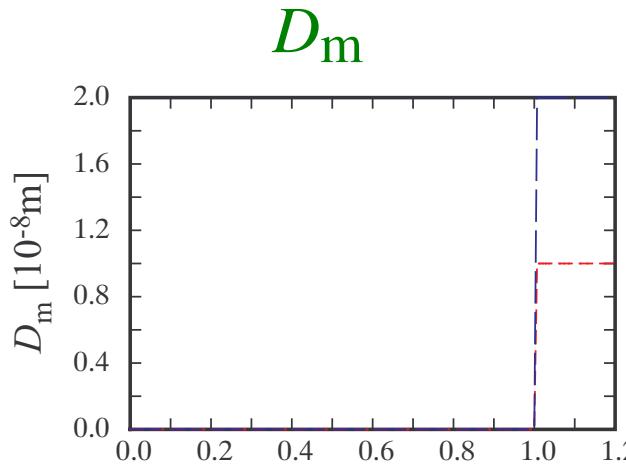
- **EC heating power:**  $P_{\text{EC}} = 0.0, 0.25, 0.5 \text{ MW}$



- EC heating in the central region leads to positive  $E_r$  and enhances the plasma temperature when the heating power is above a threshold.

# Dependence on Magnetic Braiding

- **Spatial diffusivity of the magnetic field line:**  $D_m = 0 \sim 2 \times 10^{-8} \text{ m}$



- Magnetic braiding affects  $E_r$ , but the density and temperature profiles are not affected so much in this case.

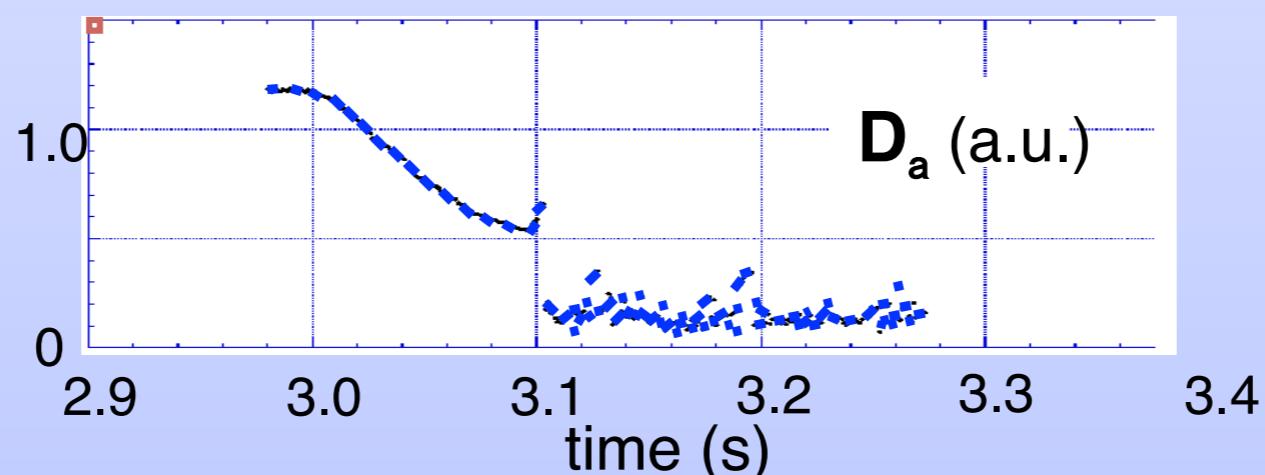
# 1D Core + 2D Peripheral

---

- \* Next talk by Yagi et al.
- \* Integrated core and SOL/Divertor simulation
  - ▶ 1D Core Transport : TOPICS
  - ▶ 2D SOL/Divertor Transport: SONIC

- \* Turbulent transport coefficient
  - ▶ Tuned CDBM model
  - ▶ LH transition

## Time evolution of $D_a$ by SONIC



# Transport + Stability + Pellet

---

- \* ELM cycle simulation

- ▶ N. Hayashi, et al., Nucl. Fusion 49 (2009) 095015

- \* Pellet induced ELM simulation

- ▶ N. Hayashi, et al., Nucl. Fusion 51 (2011) 103030

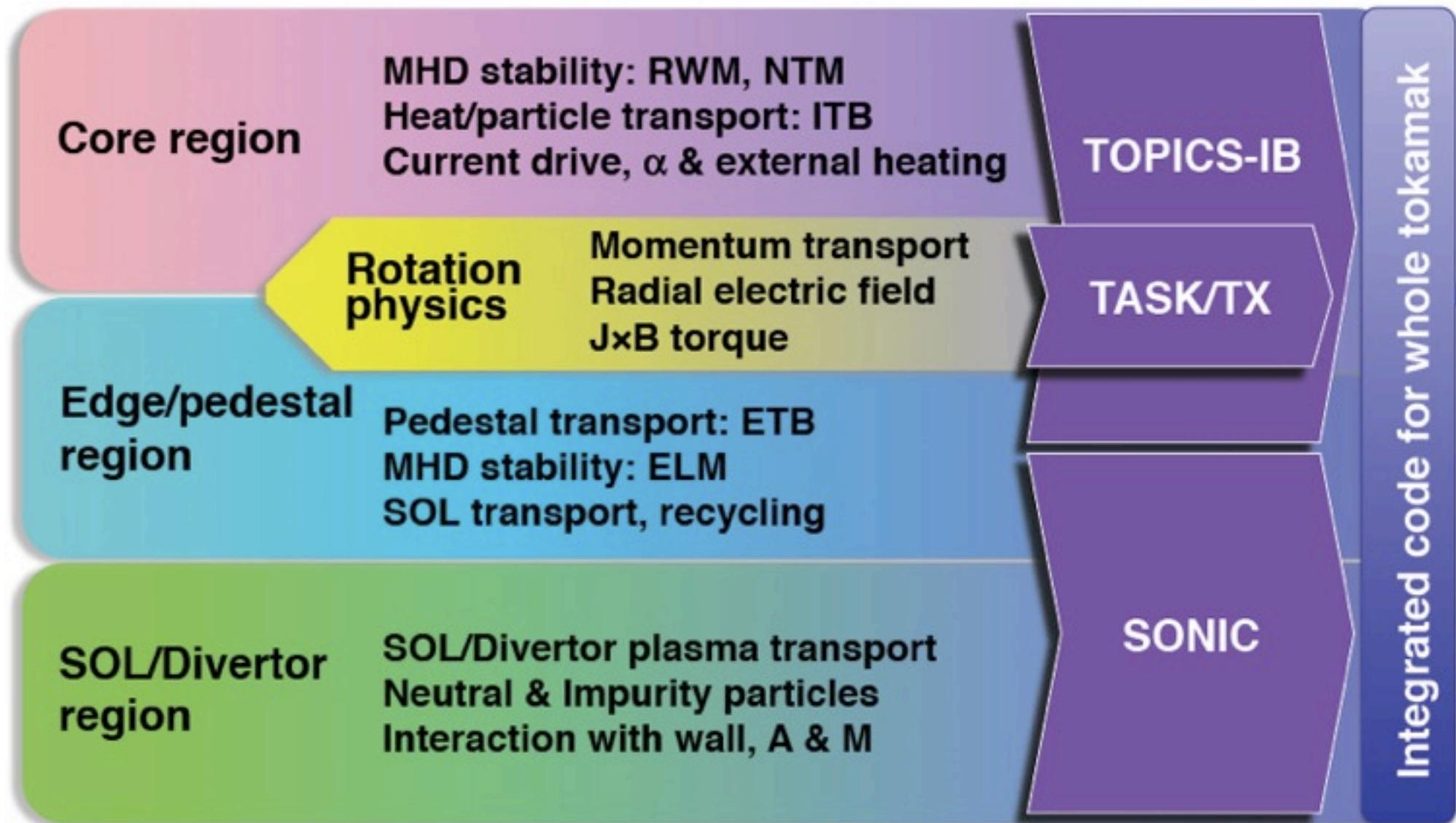
- \* Integrated modeling

- ▶ 1D transport
  - ▶ MHD stability
  - ▶ SOL-Diverter point model
  - ▶ Neutral model
  - ▶ Ablated Pellet model
- TOPICS
- MARG2D
- D5PM
- 2D Monte Carlo
- APLEX

- ▶ Comparison with experimental data
  - ▶ Benchmark test with JINTRACK

# Development of Integrated models

*Integrated modeling in JT-60*



# ELM energy loss and cycle with TOPICS-IB

## ELM version of TOPICS-IB

1.5D core transport  
( TOPICS )

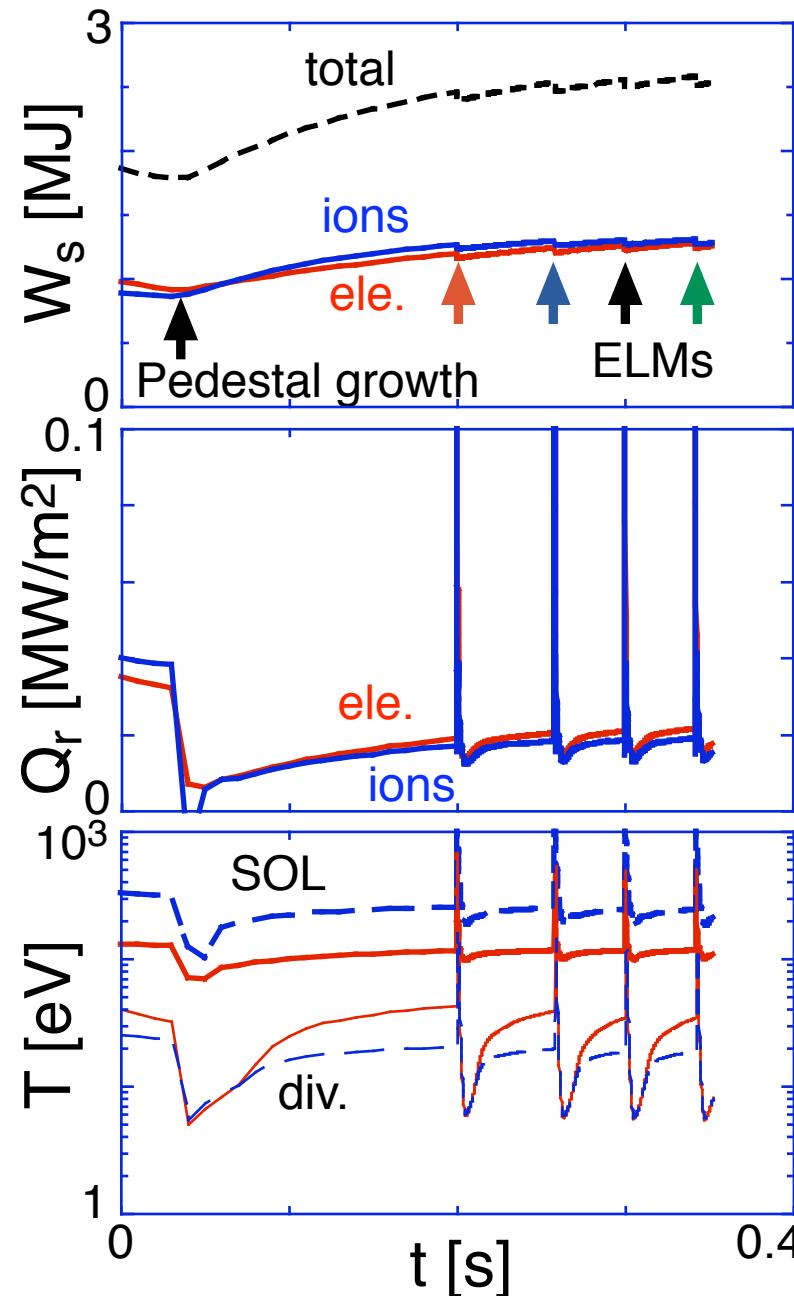
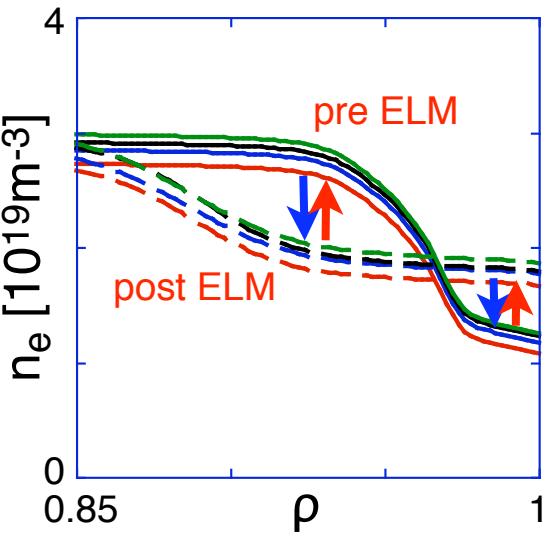
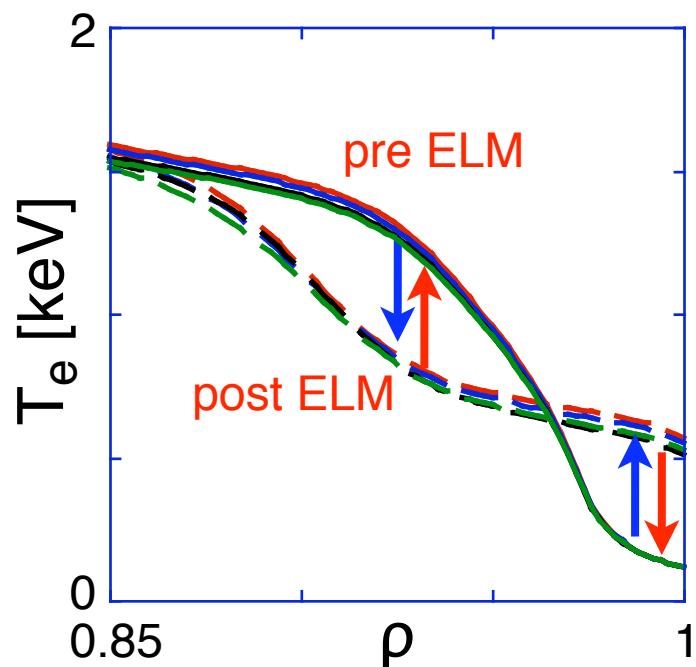
Core neutrals  
(2D Monte-Carlo)

ELM model

Linear MHD stability  
( MARG2D )

SOL-divertor  
(D5PM)

Neutral  
model



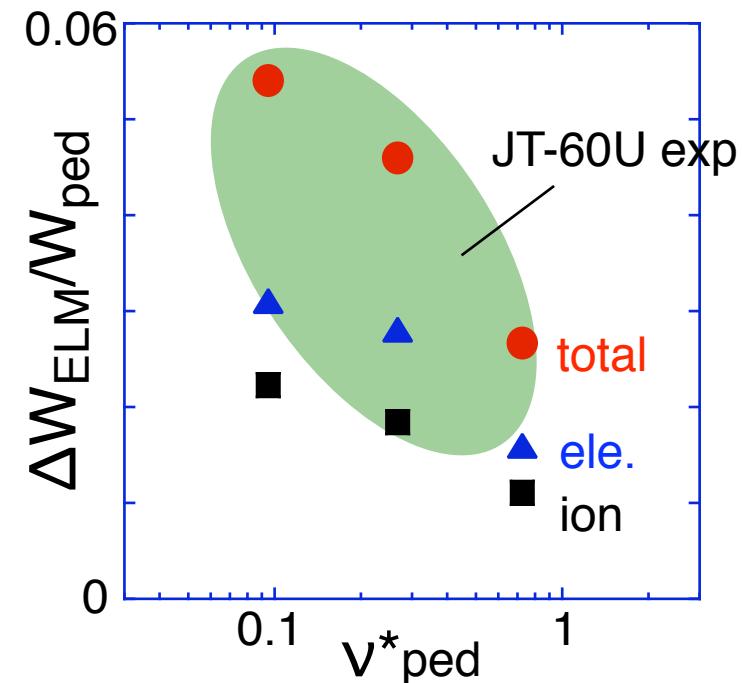
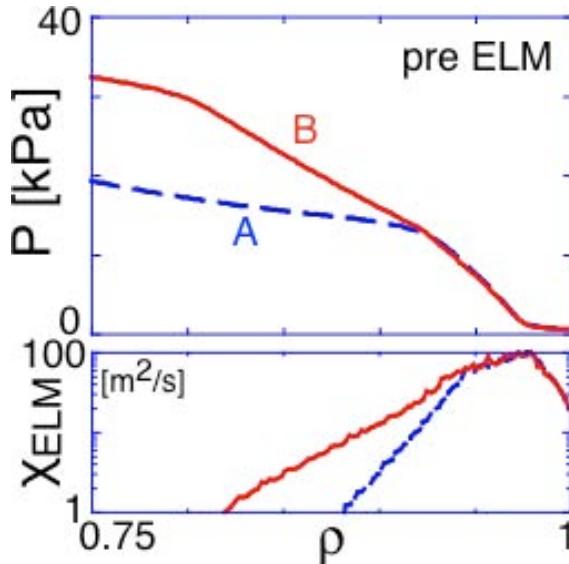
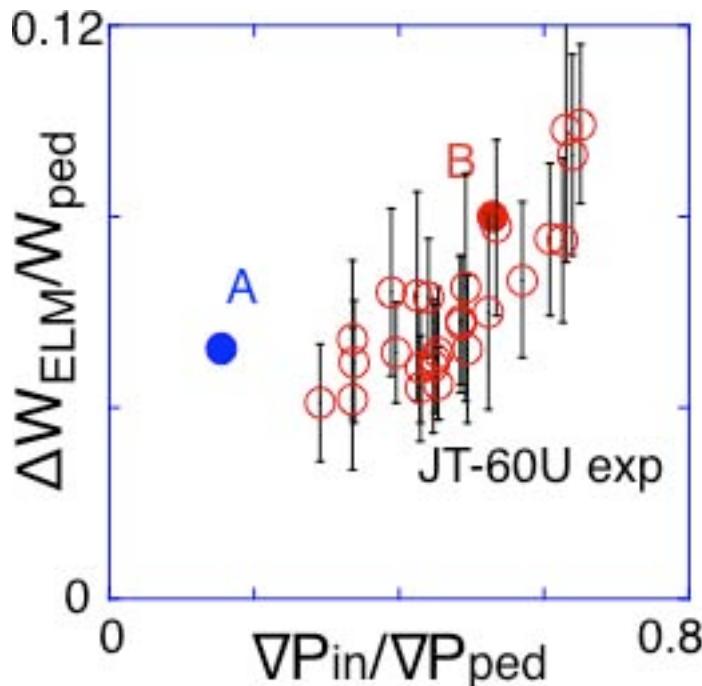
# Mechanism of ELM energy loss

(1) Collisionality dependence caused by bootstrap current, SOL parallel transport and equipartition effect

Hayashi, JPCS08

(2) Steep pressure gradient just inside the top of pedestal enhances the loss.

Hayashi, NF09

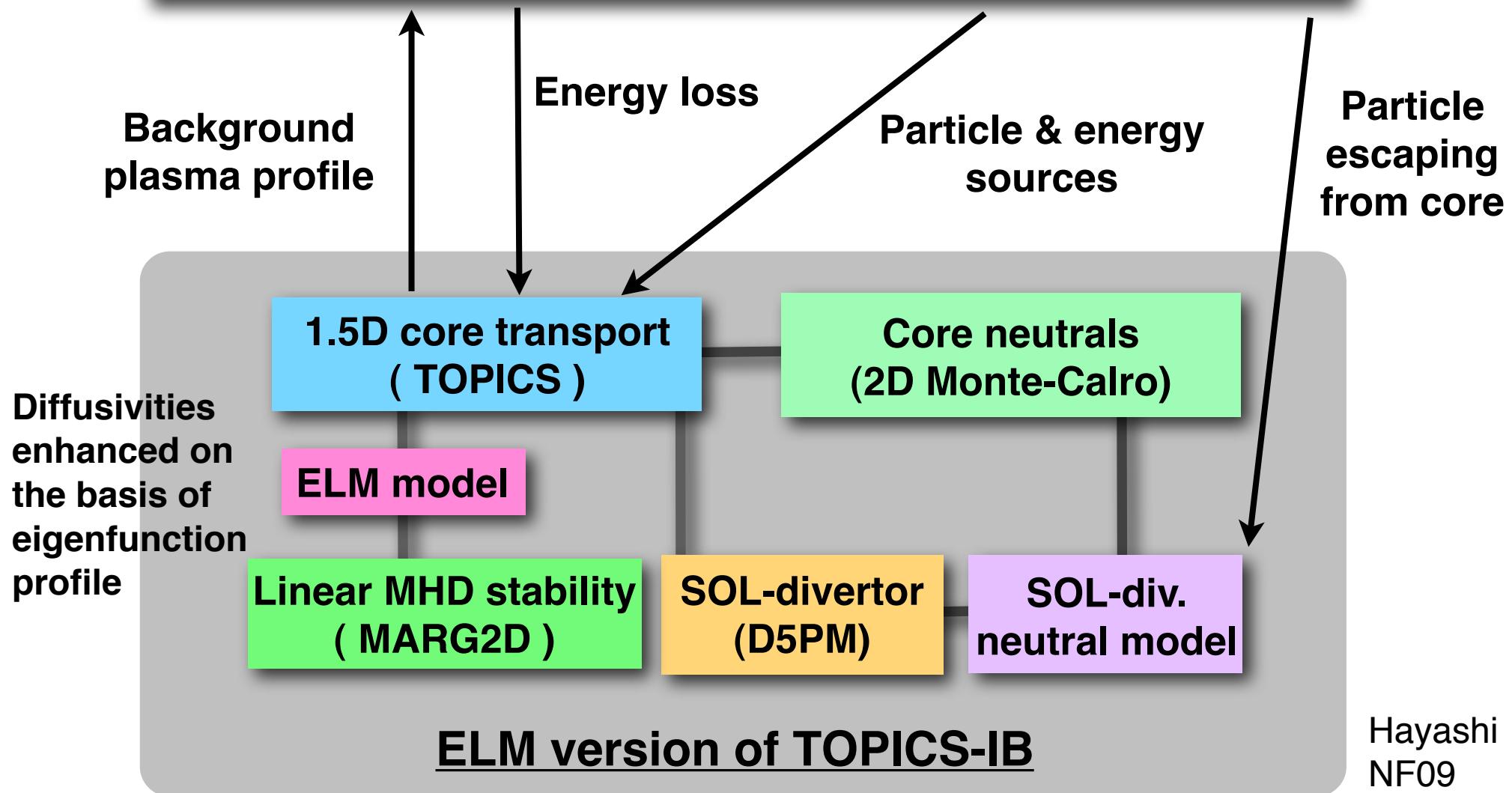


Steep pressure gradient broadens eigenfunction profile.

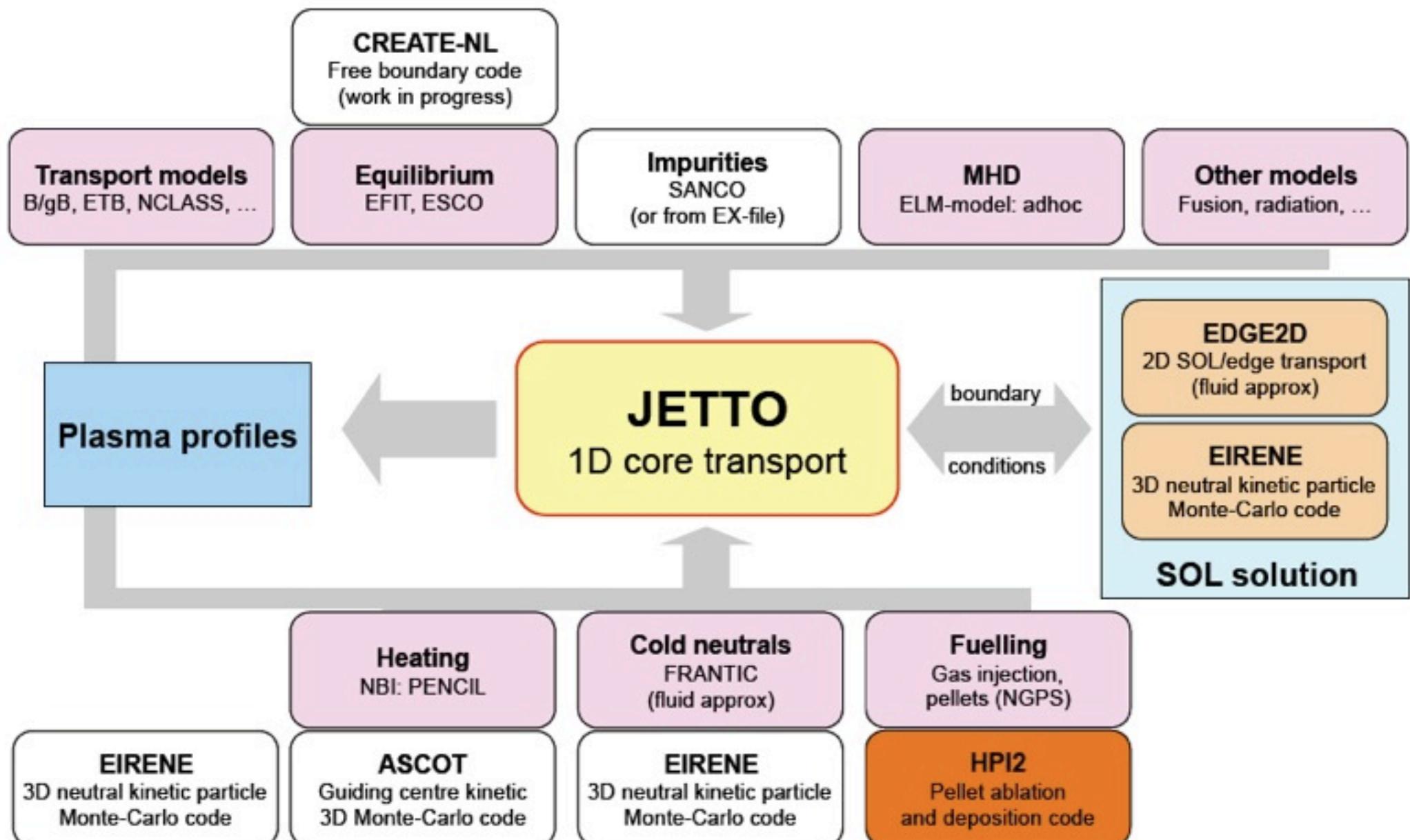
# TOPICS-IB to study pellet triggered ELM

## Ablated PeLlet with ExB drift (APLEX) model

Ablation, energy absorption, ExB drift, homogenization

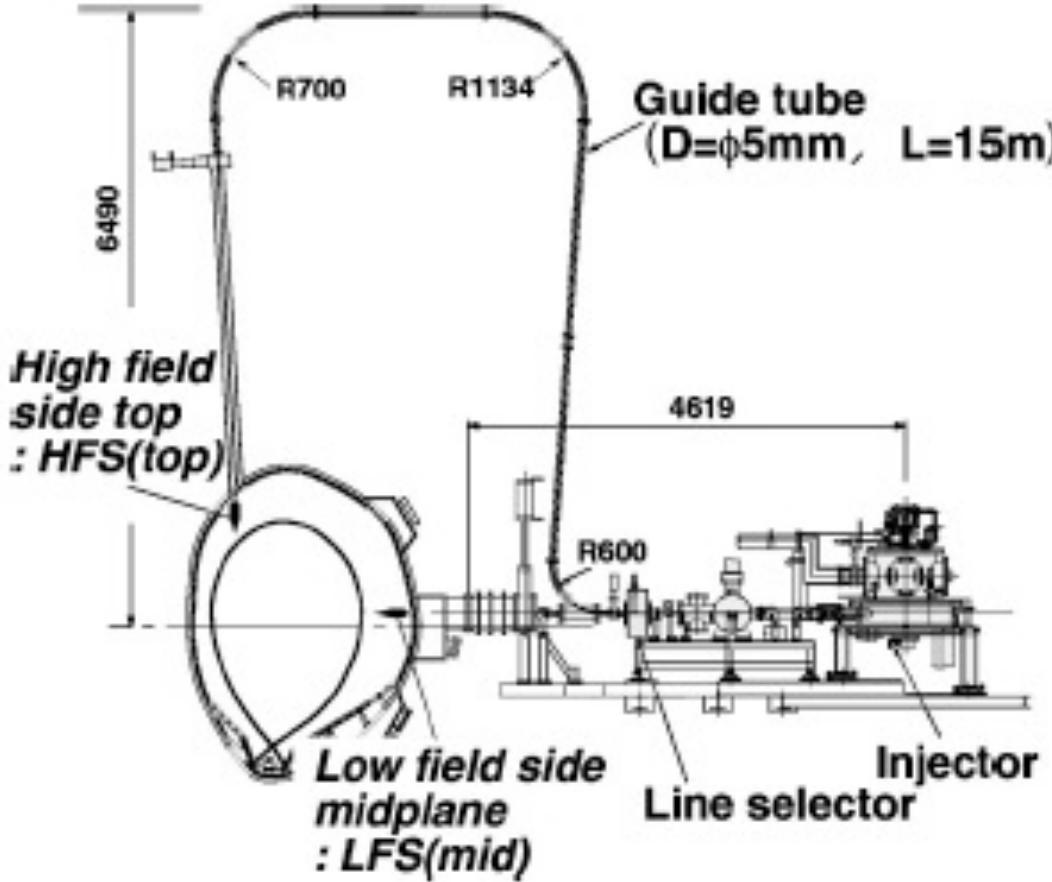


# JINTRAC to study pellet triggered ELM



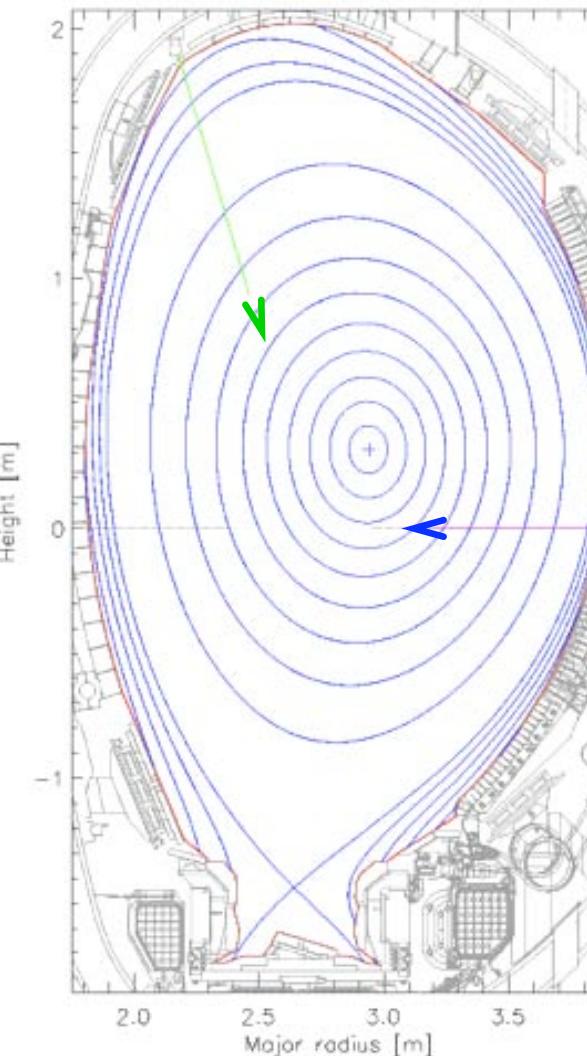
# Pellet injection in JT-60U and JET

JT-60U



JET

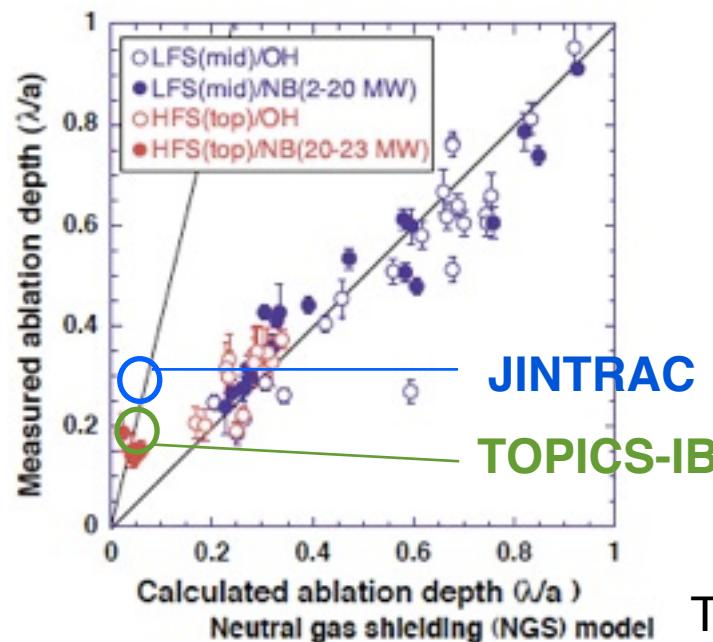
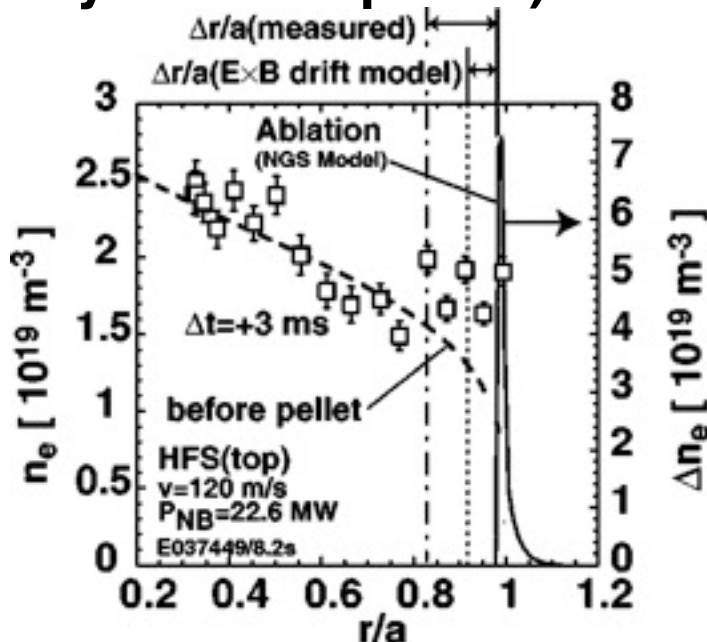
Vertical high field side injection : VHFS



Low field side injection : LFS

# Comparison with experiment: JT-60U

## Experiment (ELMy H-mode phase):



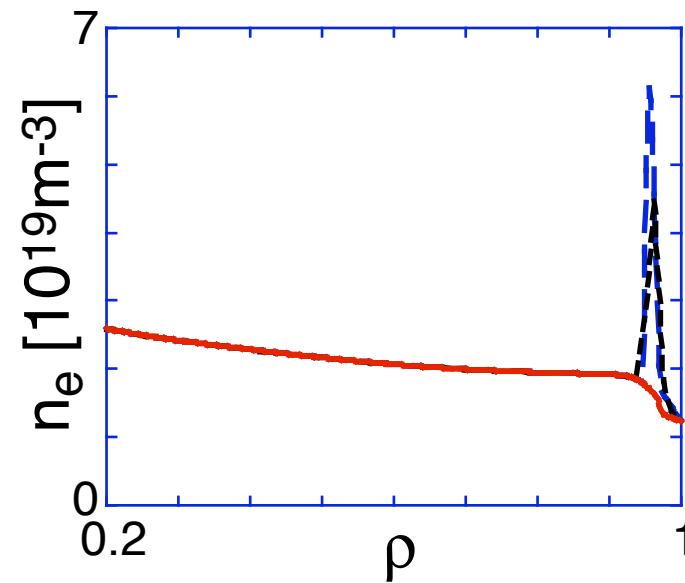
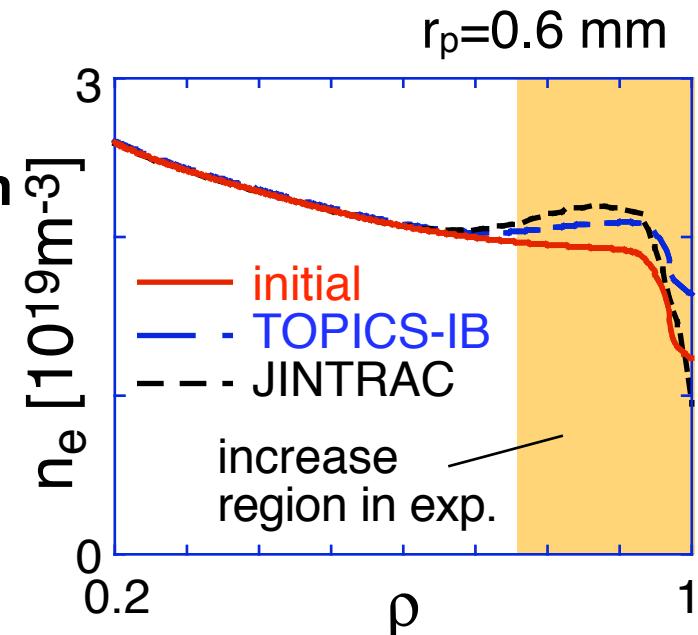
## Modeling:

- Time-dependent interaction between pellet and background plasma ON
- ExB drift ON

Penetration depth  
~ 0.2 on  $\rho$   
in TOPICS-IB  
~ 0.3 in JINTRAC

- Time-dependent interaction OFF
- ExB drift OFF

Penetration depth  
~ 0.05 on  $\rho$  in both  
codes



# Comparison with experiment: JET

**Experiment  
(L-mode phase):**

Penetration depth  $\sim 0.3$

**Modeling:**

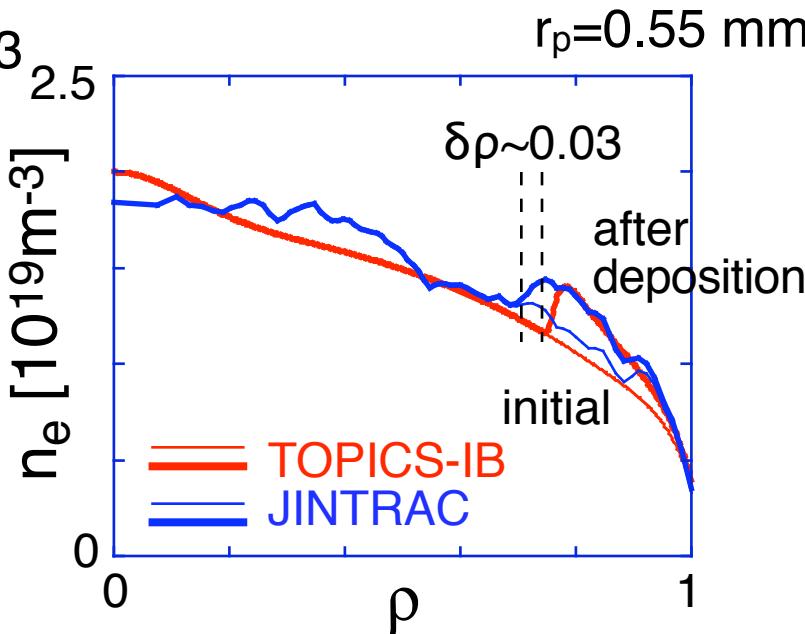
- Time-dependent interaction ON
- ExB drift ON

Approximated initial profile used in TOPICS-IB

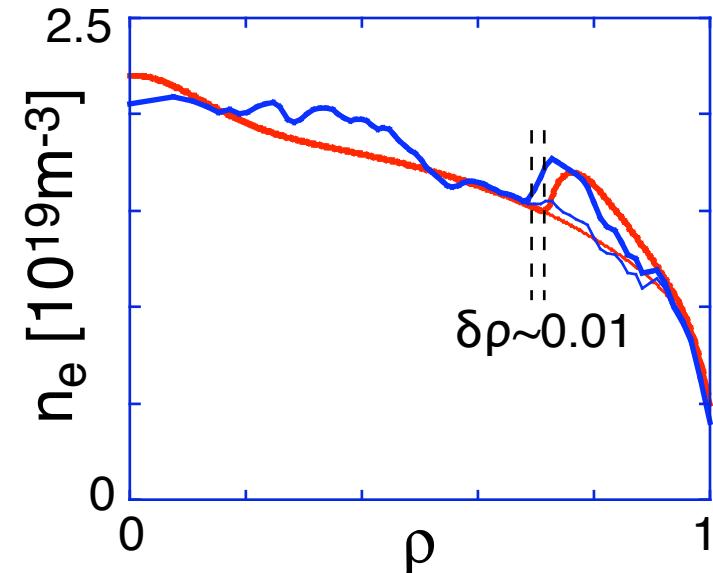
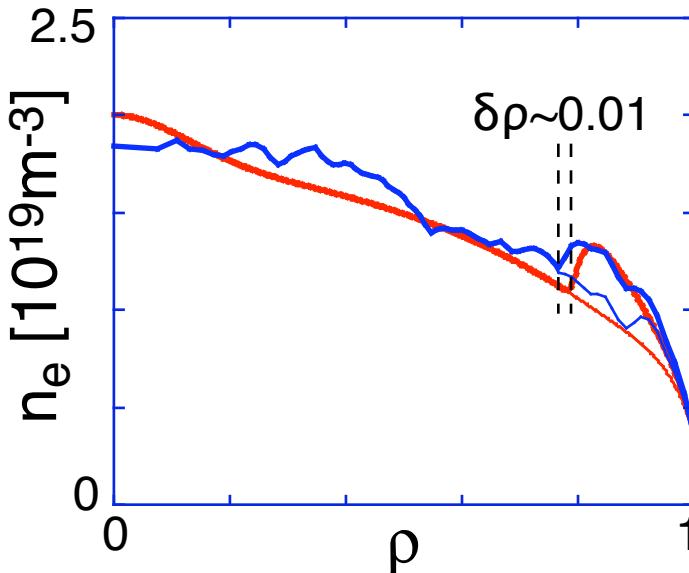
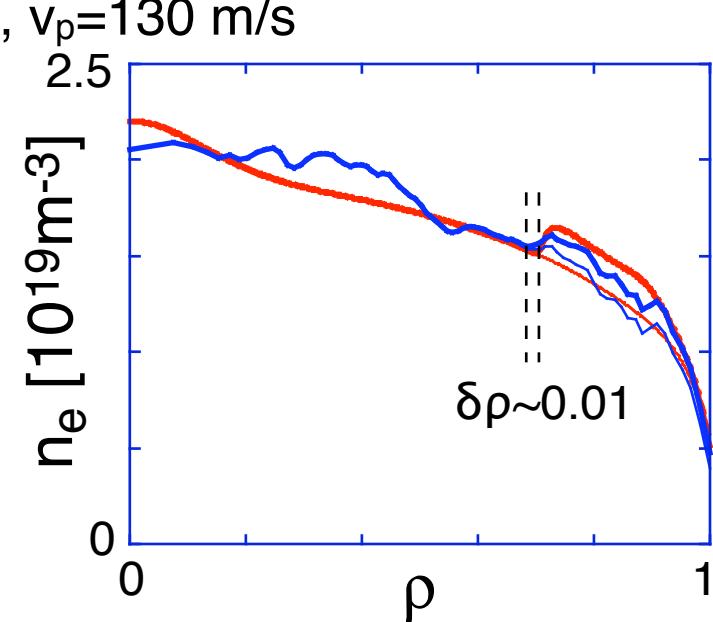
- Time-dependent interaction OFF
- ExB drift OFF

Penetration depth is slightly shorter in TOPICS-IB than JINTRAC.

**Vertical (VHFS) pellet**



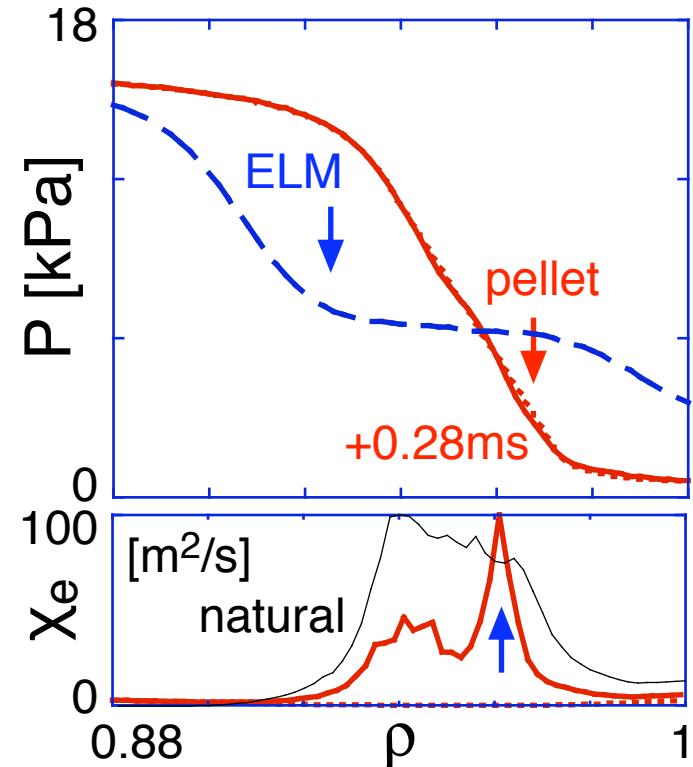
**LFS pellet**



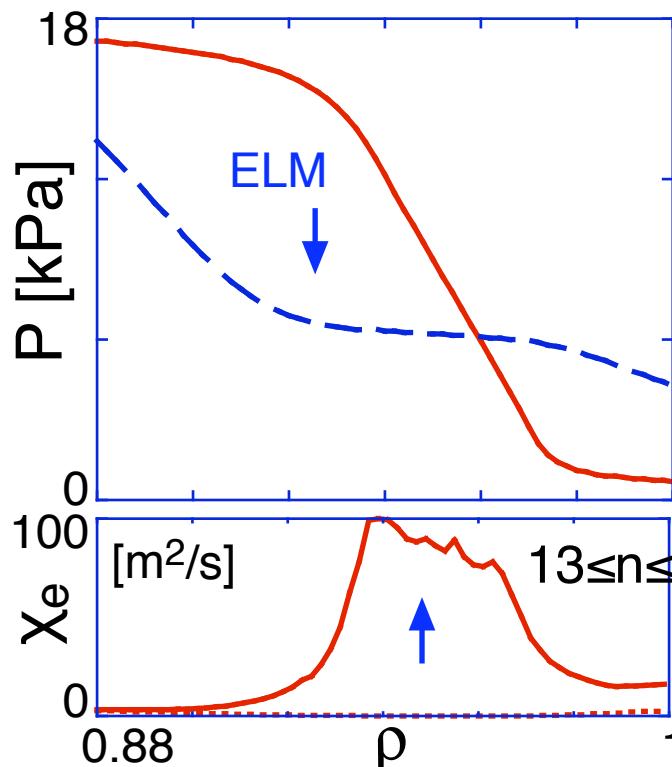
# ELM induced by pellet energy absorption

- Stabilities of  $n=1-50$  modes are examined by MARG2D.
- $n > 40$  modes become unstable in the pedestal region and their eigenfunction profiles are narrower than those in a natural ELM.
- ELM energy loss is less than half of a natural ELM.  
-> Agree with experimental observations [Kocsis NF07]

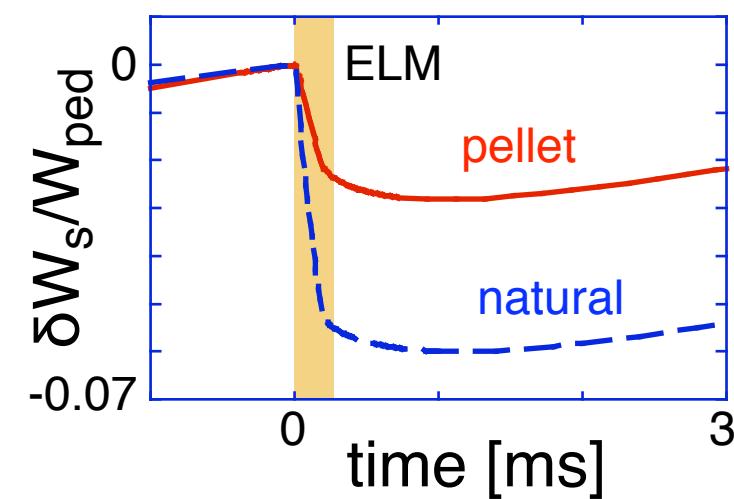
## Pellet induced ELM



## Natural ELM



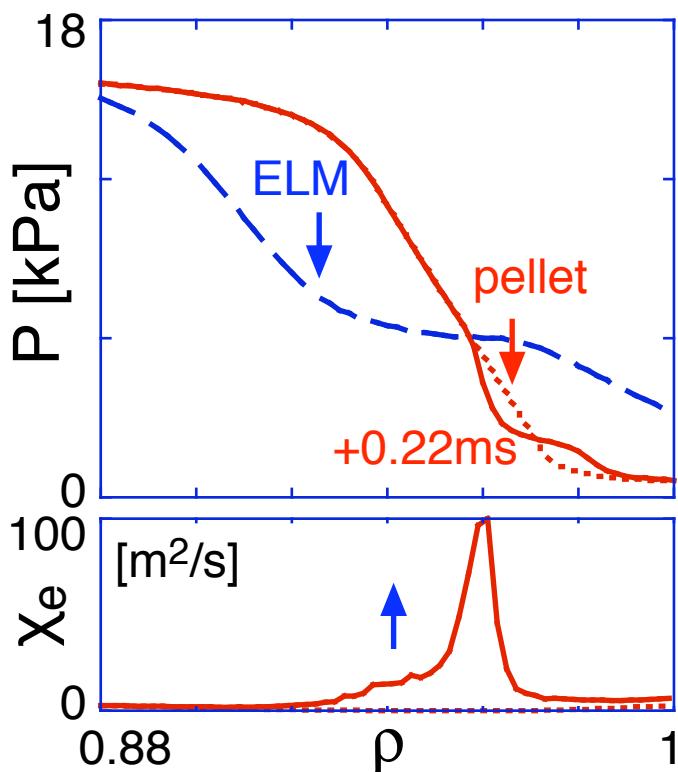
## TOPICS-IB results



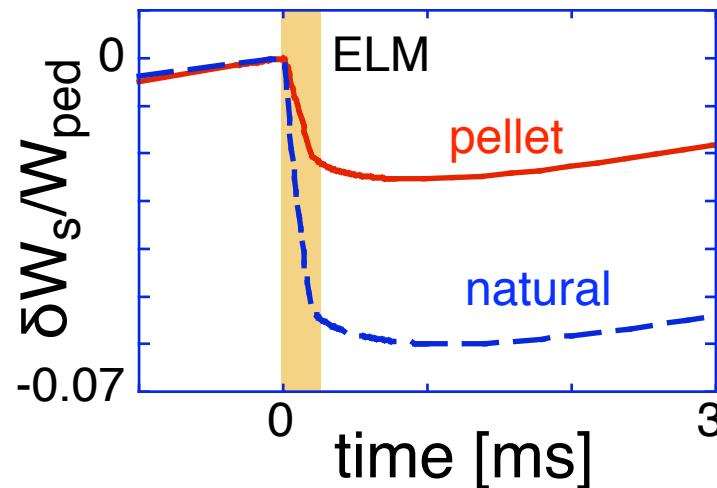
$13 \leq n \leq 28$  unstable  
based on eigenfunction profile in MARG2D

# ELM induced by pellet energy absorption and transport enhancement

- Pellet with both energy absorption and transport enhancement effects makes the background pressure perturbation even stronger.
- Either effect triggers high-n modes. (transport enhancement in VHFS pellet and energy absorption in LFS pellet)
- ELM energy loss is less than half of a natural ELM.



VHFS pellet case with TOPICS-IB



# Summary

---

- \* Various activities on integrated modeling of toroidal plasmas are going on, and new results are coming in.
- \* Data interface for ITER integrated modeling is under discussion, but that for peripheral plasma seems immature.
- \* TASK/TX describes the plasma rotation and radial electric field in edge region self-consistently.
- \* TOPICS + SONIC simulation successfully reproduced the LH transition.
- \* TOPICS + MARG2D + APLEX simulation clarified the mechanism of pellet-induced ELM.
- \* Works in progress in TASK
  - Full 2D transport modeling (P1-9 Seto) : Extension of TASK/TX
  - Kinetic integrated modeling based on 3D distribution function (velocity 2D, position 1D) TASK/FP